

MENU

SEARCH

INDEX

DETAIL

1/1



JAPANESE PATENT OFFICE

## PATENT ABSTRACTS OF JAPAN

(11)Publication number: 08305568

(43)Date of publication of application: 22. 11. 1996

(51) Int. Cl.

G06F 9/38  
G06F 7/00

(21)Application number: 07106624

(71)Applicant:

MATSUSHITA ELECTRIC IND CO LTD

(22)Date of filing: 28.04.1995

(72)Inventor:

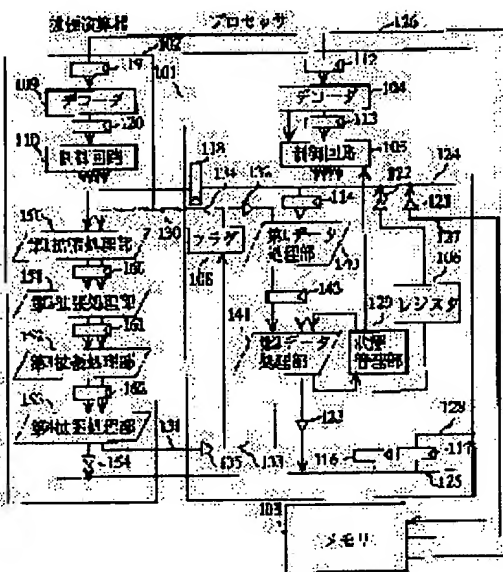
MORIKAWA TORU  
HIGAKI NOBUO  
MIYAJI SHINYA

## (54) INFORMATION PROCESSING UNIT

## (57) Abstract:

**PURPOSE:** To provide the information processing unit in which an execution time is not increased when an extension arithmetic unit carries out an extension arithmetic instruction using data from a conventional register of a processor as a source and a destination.

**CONSTITUTION:** A processor 101 decodes an extension arithmetic instruction for an extension arithmetic unit 102 and gives required operand data to the extension arithmetic unit 102 from a conventional register 106 via a data processing section input bus 124. The extension arithmetic unit 102 uses the received operand data to execute an extension arithmetic operation and gives the result of arithmetic operation to the processor 101 via a data processing section output bus 125. The processor 101 writes back the received data to the conventional register 106. Thus, the execution time can be shortened.



## LEGAL STATUS

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平8-305568

(43)公開日 平成8年(1996)11月22日

(51)Int.Cl.<sup>6</sup>

G 0 6 F 9/38  
7/00

識別記号

3 7 0

庁内整理番号

8323-5E

F I

G 0 6 F 9/38  
7/00

3 7 0 X  
U

技術表示箇所

審査請求 未請求 請求項の数11 O L (全 25 頁)

(21)出願番号 特願平7-106624

(22)出願日 平成7年(1995)4月28日

(71)出願人 000005821

松下電器産業株式会社  
大阪府門真市大字門真1006番地

(72)発明者 森河 徹

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(72)発明者 桧垣 信生

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(72)発明者 宮地 信哉

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

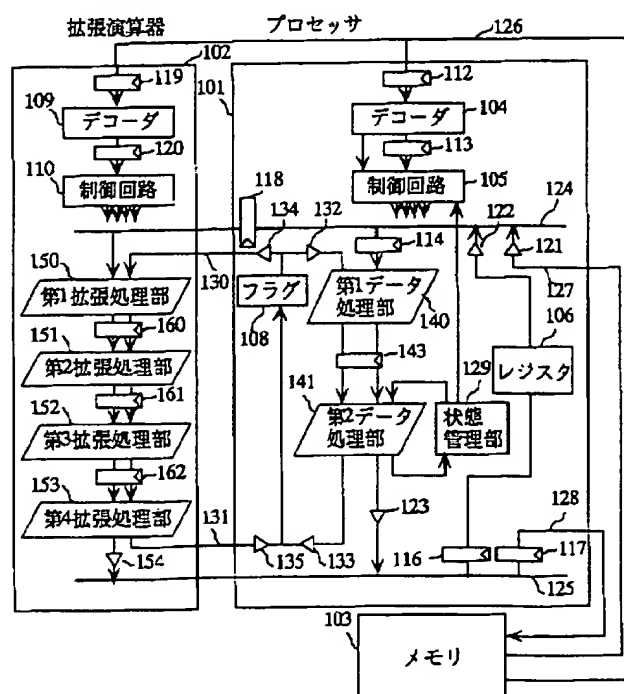
(74)代理人 弁理士 中島 司朗

(54)【発明の名称】 情報処理装置

(57)【要約】

【目的】 拡張演算器がプロセッサの汎用レジスタのデータをソース、ディスチネーションとする拡張演算命令を行う場合に、実行時間が増大しない情報処理装置を提供することを目的とする。

【構成】 プロセッサ101は、拡張演算器102用の拡張演算命令を解釈し、必要なオペランドデータを汎用レジスタ106からデータ処理部入力バス124を介して供給する。拡張演算器102は、供給されたオペランドデータを用いて拡張演算を実行し、演算結果をデータ処理部出力バス125を介してプロセッサ101に送る。プロセッサ101では、送られたデータを汎用レジスタ106に書き戻す。これにより実行時間を短縮することができる。



1

## 【特許請求の範囲】

【請求項 1】 メモリ中の命令に従ってデータを処理する主たるデータ処理装置と従たる拡張処理装置とからなる情報処理装置であって、

データ処理装置は、

複数のデータ保持領域を有するデータ保持手段と、

メモリ中の命令を取り出す命令取り出し手段と、

取り出された命令を解読し、拡張処理装置で実行すべき拡張演算命令を検出する命令検出手段と、

検出された拡張演算命令のオペランドでソースとして指定されたデータ保持領域と、デスティネーションとして指定されたデータ保持領域とを判別するオペランド判別手段と、

判別されたソースのデータ保持領域のデータを拡張処理装置に供給するデータ供給手段と、

拡張処理装置から伝送される演算結果を、判別されたデスティネーションのデータ保持領域に格納するデータ格納手段とを備え、

拡張処理装置は、

取り出された命令を解読して拡張演算命令を検出する拡張命令解読手段と、

データ供給手段から供給されるデータを用いて、拡張命令解読手段に検出された拡張演算命令を実行する拡張実行手段とを備え、

データ処理装置と拡張処理装置とは、メモリから取り出された命令を命令検出手段と拡張命令解読手段とに伝送する命令バス、データ供給手段の供給データを拡張実行手段に伝送する第 1 のバス、及び、拡張実行手段の演算結果をデータ格納手段に伝送する第 2 のバスにより接続されていることを特徴とする情報処理装置。

【請求項 2】 前記情報処理装置は、さらに複数の前記拡張処理装置を備え、

各拡張処理装置は、演算種類が互いに異なる拡張演算命令を実行し、

前記命令バスは、メモリから取り出された命令を各拡張命令解読手段に伝送し、

前記第 1 のバスは、データ供給手段の供給データを各拡張実行手段に伝送し、

前記第 2 のバスは、各拡張実行手段の演算結果をデータ格納手段に伝送することを特徴とする請求項 1 記載の情報処理装置。

【請求項 3】 前記データ処理装置は、さらに、演算結果の状態を表す複数のフラグを保持するフラグレジスタを備え、

前記データ供給手段は、ソースのデータ保持領域のデータ供給とともに、フラグレジスタのフラグを拡張処理装置に供給し、

前記データ格納手段は、演算結果をデスティネーションのデータ保持領域に格納するとともに、拡張処理装置からの新たなフラグをフラグレジスタに格納し、

2

前記拡張実行手段は、データ供給手段から供給されるデータとともにフラグを用いて拡張演算命令を実行し、当該演算の状態を表す新たなフラグをデータ格納手段に出力し、

データ処理装置と拡張処理装置とは、さらに、データ供給手段により供給されるフラグを拡張実行手段に伝送する第 1 のフラグバス、及び、拡張実行手段からの新たなフラグをフラグレジスタに伝送する第 2 のフラグバスにより接続されていることを特徴とする請求項 1 記載の情報処理装置、

【請求項 4】 前記データ処理装置は、少なくとも命令フェッチステージと、命令解読ステージと命令実行ステージとを含むパイプライン処理を行い、さらに、拡張実行手段における拡張演算の実行サイクル数に、命令実行ステージのサイクル数を一致させるステージ状態管理手段と、

を備え、

前記拡張処理装置は、データ処理装置に同期して動作することを特徴とする請求項 2 記載の情報処理装置。

【請求項 5】 前記拡張実行手段は、拡張演算の実行に際して、命令実行の最後のサイクルになったときに、ステージ状態管理手段に拡張演算の実行終了を通知し、前記ステージ状態管理手段は、拡張演算の実行終了通知を受けると、受けた時点の実行サイクルで命令実行ステージを終了させることを特徴とする請求項 4 記載の情報処理装置。

【請求項 6】 前記データ処理装置は、検出手段により検出された拡張演算命令に応じて、データ処理装置の実行ステージのサイクル数を決定する決定手段を備え、

前記ステージ状態管理手段は、決定されたサイクル数の間命令実行ステージを継続させることを特徴とする請求項 4 記載の情報処理装置。

【請求項 7】 前記命令検出手段は、データ処理装置から拡張処理装置へのデータ転送を指示する第 1 の転送命令を検出し、

前記オペランド判別手段は、検出された第 1 の転送命令のオペランドでソースとしての指定されたデータ保持領域を判別し、

前記データ供給手段は、判別された第 1 の転送命令のソースのデータ保持領域のデータを第 1 のバスに出力させ、

前記拡張処理装置は、さらに、データを保持する入力バッファを備え、

前記拡張命令解読手段は、前記転送命令を検出した場合、第 1 のバスからのデータを入力バッファに格納し、前記拡張命令実行手段は、3 つ以上のオペランドを要する拡張演算命令の実行に入力バッファのデータを用いることを特徴とする請求項 1、3 または 4 記載の情報処理装置。

【請求項 8】 前記情報処理装置は、さらに、データを記憶する領域を有する退避バッファを備え、  
 前記拡張命令解読手段は、割り込み処理からの復帰命令を検出し、  
 前記データ処理装置は、割り込み要求を受け付けると、割り込み信号を拡張処理装置に出力し、  
 前記拡張処理装置は、さらに割り込み信号を受けると、前記入力バッファのデータを退避バッファに退避させる退避手段と、  
 割り込み処理からの復帰命令が検出されると、退避バッファのデータを前記入力バッファに復帰させる復帰手段とを備えることを特徴とする請求項 7 記載の情報処理装置。

【請求項 9】 前記拡張処理装置は、さらに、拡張実行手段の演算結果の一部を保持する出力バッファを備え、  
 前記拡張命令解読手段は、拡張処理装置からデータ処理装置へのデータ転送を指示する第 2 の転送命令を解読した場合、出力バッファのデータを第 2 のバスに出力させ、  
 前記命令検出手段は、前記第 2 の転送命令を検出し、  
 前記オペランド判別手段は、検出された第 2 の転送命令のオペランドでディスチネーションとして指定されたデータ保持領域を判別し、  
 前記データ格納手段は、第 2 の転送命令のディスチネーションのデータ保持領域に第 2 のバスからのデータを格納することを特徴とする請求項 1、3、4、または 7 記載の情報処理装置。

【請求項 10】 前記情報処理装置は、さらに、複数のデータを記憶する領域を有する退避バッファを備え、  
 前記データ処理装置は、割り込み要求を受け付けると、割り込み信号を拡張処理装置に出力し、  
 前記拡張処理装置は、さらに割り込み信号を受けると、前記入力バッファおよび出力バッファのデータを退避バッファに退避させる退避手段と、  
 割り込み処理からの復帰命令が検出されると、退避バッファのデータを前記入力バッファ及び出力バッファに復帰させる復帰手段と、  
 を備え、  
 前記拡張命令解読手段は、割り込み処理からの復帰命令を検出すると、その旨を復帰手段に通知することを特徴とする請求項 9 記載の情報処理装置。

【請求項 11】 前記拡張命令解読手段は、タスクを識別するタスク番号を伴うタスク切り替え命令を解読し、  
 前記拡張処理装置は、さらに、複数の前記入力バッファと、同数の前記出力バッファとを対にして備え、  
 解読されたタスク切り替え命令に示されるタスク番号に対応させた 1 対の入力バッファと出力バッファに対するデータ入出力を許可し、他の入力バッファと出力バッファに対するデータ入出力を禁止するタスク管理手段を備えることを特徴とする請求項 9 記載の情報処理装置。

# 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は、メモリ中の命令に従ってデータを処理する主たるデータ処理装置と従たる拡張処理装置とからなる情報処理装置に関する。

## 【0002】

【従来の技術】 近年、データ処理装置（以下、プロセッサと記す）とともに、用途に応じた特別な演算を高速に実行する拡張演算器を設けて処理能力の向上が図られている。この拡張演算器の実装方式には、プロセッサに内蔵する方式や、必要な場合のみ接続して実装する接続方式などがある。内蔵方式では、拡張演算器を用いない場合に、コストの増大と消費電力の増大とを引き起こすというデメリットがあるので、性能を劣化させずにプロセッサと拡張演算器とを接続する方式が求められている。

【0003】 従来のプロセッサと拡張演算器を接続した情報処理装置として、プロセッサが拡張演算器の命令を解読して、拡張演算器にコマンドを通知することによって、拡張演算を実行させる方式がある。この方式では、プロセッサからのコマンド転送のオーバーヘッドが生じることが原因で拡張演算器の処理時間が長くなるという欠点がある。

【0004】 また、特開平 1 - 2 4 0 9 3 2 号公報では、拡張演算器でもパイプライン処理を行うプロセッサと同じ命令を同期を取って解読することによりコマンド転送及びコマンド解読に要する時間を隠す情報処理装置が考案されている。

## 【0005】

【発明が解決しようとする課題】 しかしながら、従来の情報処理装置によれば、プロセッサ内の汎用レジスタが持つデータを用いて、拡張演算器が処理を行う場合や、拡張演算器が処理を行ったデータをプロセッサが持つ汎用レジスタに格納する場合には、データを一旦メモリに保持させて、データを転送するため、実行時間が増大し性能が劣化するという課題を有していた。

【0006】 また、プロセッサのフラグ格納レジスタが持つフラグ情報を用いて、拡張演算器が処理を行う場合、又は、拡張演算器が処理を行ったフラグ情報をプロセッサが持つフラグ格納レジスタに格納する場合、フラグ情報を一旦メモリに保持することによりフラグ情報を転送するため、実行時間が増大し性能が劣化するという課題を有していた。

【0007】 さらに、プロセッサの汎用レジスタが持つデータ、拡張演算器が処理を行ったフラグ情報を、プロセッサが持つ汎用レジスタ、フラグ格納レジスタに格納する場合に、プロセッサがデータ、又は、フラグ情報を汎用レジスタ、フラグ格納レジスタに格納を完了するまで、パイプライン処理を停止させる必要があるため、実行時間が増大し性能が劣化するという課題を有していた。

【0008】さらに、拡張演算器が処理中に割り込み処理が生じた場合、処理中のデータを保持することができないので、割り込み処理後に再度拡張演算器が処理するため、実行時間が増大し性能が劣化するという課題を有していた。また、拡張演算器が処理中にタスク切り換え処理が生じた場合、処理中のデータを保持することができないので、再度タスク切り替え前のタスク処理時に、拡張演算器が再度処理するため、実行時間が増大し性能が劣化するという課題を有していた。

【0009】上記問題点に鑑み本発明は、プロセッサの汎用レジスタが持つデータを用いて、拡張演算器が処理を行う場合、又は、拡張演算器が処理を行ったデータをプロセッサが持つ汎用レジスタに格納する場合において、実行時間が増大しない情報処理装置を提供することを目的とする。さらに本発明は、プロセッサの汎用レジスタが持つデータを用いて、拡張演算器が処理を行う場合、又は、拡張演算器が処理を行ったデータをプロセッサが持つ汎用レジスタに格納する場合、高速に処理し、拡張演算器を必要としない場合、回路規模の増大によるコスト上昇と消費電力の増大とを抑止する情報処理装置を提供することを目的とする。

【0010】また本発明は、プロセッサの汎用レジスタが持つデータを用いて、拡張演算器が処理を行う、又は、拡張演算器が処理を行ったデータをプロセッサが持つ汎用レジスタに格納する場合において、プロセッサのパイプラインを乱すことなく高速に実行する情報処理装置を提供することを目的とする。さらに本発明は、プロセッサのフラグ格納レジスタが持つフラグ情報を用いて、拡張演算器が処理を行う場合、又は、拡張演算器が処理を行ったフラグ情報をプロセッサが持つフラグ格納レジスタに格納する場合、実行時間が増大し性能が劣化しない情報処理装置を提供することを目的とする。

【0011】さらに本発明は、拡張演算器が複数のオペランドを処理する場合、チップ面積の増大によるコスト上昇と消費電力の増大とを抑止する情報処理装置を提供することを目的とする。また本発明は、拡張演算器が処理中に割り込み処理が生じた場合、実行時間が増大と性能が劣化とを抑止する情報処理装置を提供することを目的とする。

【0012】さらに本発明は、拡張演算器が処理中にタスク切り換え処理が生じた場合、実行時間が増大と性能が劣化とを抑止する情報処理装置を提供することを目的とする。

【0013】

【課題を解決するための手段】上記課題を解決するため、請求項1の発明は、メモリ中の命令に従ってデータを処理する主たるデータ処理装置と従たる拡張処理装置とからなる情報処理装置であって、データ処理装置は、複数のデータ保持領域を有するデータ保持手段と、メモリ中の命令を取り出す命令取り出し手段と、取り出され

た命令を解読し、拡張処理装置で実行すべき拡張演算命令を検出する命令検出手段と、検出された拡張演算命令のオペランドで指定されたソースのデータ保持領域とデスティネーションのデータ保持領域とを判別するオペランド判別手段と、判別されたソースのデータ保持領域のデータを拡張処理装置に供給するデータ供給手段と、拡張処理装置から伝送される演算結果を、判別されたデスティネーションのデータ保持領域に格納するデータ格納手段とを備え、拡張処理装置は、取り出された命令を解読して拡張演算命令を検出する拡張命令解読手段と、データ供給手段から供給されるデータを用いて、拡張命令解読手段に検出された拡張演算命令を実行する拡張実行手段とを備え、データ処理装置と拡張処理装置とは、メモリから取り出された命令を命令検出手段と拡張命令解読手段とに伝送する命令バス、データ供給手段の供給データを拡張実行手段に伝送する第1のバス、及び、拡張実行手段の演算結果をデータ格納手段に伝送する第2のバスにより接続されている。

【0014】請求項2の発明は、請求項1に加えて、演算種類が互いに異なる拡張演算命令を実行する複数の前記拡張処理装置を備えている。請求項3の発明は、請求項1の情報処理装置において前記データ処理装置が、さらに、演算結果の状態を表す複数のフラグを保持するフラグレジスタを備え、前記データ供給手段は、ソースレジスタのデータ供給とともに、フラグレジスタのフラグを拡張処理装置に供給し、前記データ格納手段は、演算結果をデスティネーションのデータ保持領域に格納するとともに、拡張処理装置からの新たなフラグをフラグレジスタに格納し、前記拡張実行手段は、データ供給手段から供給されるデータとともにフラグを用いて拡張演算命令を実行し、当該演算の状態を表す新たなフラグをデータ格納手段に出力し、データ処理装置と拡張処理装置とは、さらに、データ供給手段により供給されるフラグを拡張実行手段に伝送する第1のフラグバス、及び、拡張実行手段からの新たなフラグをフラグレジスタに伝送する第2のフラグバスにより接続されている。

【0015】請求項4の発明は、請求項2の情報処理装置において前記データ処理装置が、少なくとも命令フェッチステージと、命令解読ステージと命令実行ステージとを含むパイプライン処理を行い、さらに、拡張実行手段における拡張演算の実行サイクル数に、命令実行ステージのサイクル数を一致させるステージ状態管理手段とを備え、前記拡張処理装置は、データ処理装置に同期して動作する。

【0016】請求項5の発明は、請求項4の情報処理装置において、前記拡張実行手段が、拡張演算の実行に際して、命令実行の最後のサイクルになったときに、ステージ状態管理手段に拡張演算の実行終了を通知し、前記ステージ状態管理手段は、拡張演算の実行終了通知を受けると、受けた時点の実行サイクルで命令実行ステージ

を終了させる。

【0017】請求項6の発明は、請求項4の情報処理装置において、前記データ処理装置が、検出手段により検出された拡張演算命令に応じて、データ処理装置の実行ステージのサイクル数を決定する決定手段を備え、前記ステージ状態管理手段は、決定されたサイクル数の間命令実行ステージを継続させる。請求項7の発明は、請求項1、3または4の情報処理装置において、前記命令検出手段が、データ処理装置から拡張処理装置へのデータ転送を指示する第1の転送命令を検出し、前記オペランド判別手段は、検出された第1の転送命令のオペランドでソースとして指定されたデータ保持領域を判別し、前記データ供給手段は、判別された第1の転送命令のソースのデータ保持領域のデータを第1のバスに出力させる。前記拡張処理装置は、さらに、データを保持する入力バッファを備え、前記拡張命令解読手段は、前記転送命令を検出した場合、第1のバスからのデータを入力バッファに格納し、前記拡張命令実行手段は、3つ以上のオペランドを要する拡張演算命令の実行に入力バッファのデータを用いる。

【0018】請求項8の発明は、請求項7の情報処理装置において、さらに、データを記憶する領域を有する退避バッファを備え、前記データ処理装置は、割り込み要求を受け付けると、割り込み信号を拡張処理装置に出力し、前記拡張命令解読手段は、割り込み処理からの復帰命令を検出し、前記拡張処理装置は、さらに割り込み信号を受けると前記入力バッファのデータを退避バッファに退避させる退避手段と、割り込み処理からの復帰命令が検出されると、退避バッファのデータを前記入力バッファに復帰させる復帰手段とを備える。

【0019】請求項9の発明は、請求項7において、前記拡張処理装置が、さらに拡張実行手段の演算結果の一部を保持する出力バッファを備え、前記拡張命令解読手段は、拡張処理装置からデータ保持手段へのデータ転送を指示する第2の転送命令を検出した場合、出力バッファのデータを第2のバスに出力し、前記命令検出手段は、前記第2の転送命令を検出し、前記オペランド判別手段は、検出された第2の転送命令のディスチネーションのデータ保持領域を判別し、前記データ格納手段は、第2の転送命令のディスチネーションのデータ保持領域に第2のバスからのデータを格納する。

【0020】請求項10の発明は、請求項9の情報処理装置において、さらに複数のデータを記憶する領域を有する退避バッファを備え、前記データ処理装置は、割り込み要求を受け付けると、割り込み信号を拡張処理装置に出力し、前記拡張処理装置は、さらに割り込み信号を受けると、前記入力バッファおよび出力バッファのデータを退避バッファに退避させる退避手段と、割り込み処理からの復帰命令が検出されると、退避バッファのデータを前記入力バッファ及び出力バッファに復帰させる復帰手段

と、を備え、前記拡張命令解読手段は、割り込み処理からの復帰命令を検出すると、その旨を復帰手段に通知する。

【0021】請求項11の発明は、請求項9の情報処理装置において、前記拡張命令解読手段が、タスクを識別するタスク番号を伴うタスク切り替え命令を解読し、前記拡張処理装置は、さらに、複数の前記入力バッファと、同数の前記出力バッファとを対にして備え、解読されたタスク切り替え命令に示されるタスク番号に対応させた1対の入力バッファと出力バッファに対するデータ入出力を許可し、他の入力バッファと出力バッファに対するデータ入出力を禁止するタスク管理手段を備える。

【0022】

【作用】上記課題を解決するため、請求項1の発明に係る情報処理装置では、データ処理装置は、拡張処理装置用の拡張演算命令を解読し、拡張処理装置が必要とするオペランドデータをデータ保持手段から第1のバスを介して供給する。拡張処理装置は、供給されたオペランドデータを用いて拡張演算を実行し、演算結果を第2のバスを介してデータ処理装置に送り返す。データ処理装置では、データ格納手段が送り返されたデータをデータ保持手段に書き戻す。これにより、拡張処理装置がデータ処理装置のデータ保持手段のデータを用いて演算する場合に、一旦メモリを介してデータ転送する必要がなくなり、データ処理装置と拡張処理装置との間で直接データの受け渡しをするので、処理能力の高い情報処理装置を実現することができる。

【0023】請求項2の発明に係る情報処理装置では、請求項1に加えて、演算種類が互いに異なる拡張演算命令を実行する複数の前記拡張処理装置を備える。したがって同時に複数の拡張処理装置を接続して、多様な拡張演算命令が混在するプログラムを高速に実行することができる。請求項3の発明に係る情報処理装置では、請求項1の情報処理装置に対して、複数のフラグもオペランドデータと同様に、データ処理装置と拡張処理装置との間で直接受け渡しをするので、より処理能力の高い情報処理装置を実現することができる。

【0024】請求項4の発明に係る情報処理装置では、請求項2の情報処理装置に対して、ステージ状態管理手段が、前記拡張処理装置の拡張演算実行ステージとデータ処理装置の実行ステージとで、実際のステージのサイクル数を合わせることで、データ処理装置のパイプライン動作を乱すことなく、拡張演算命令の実行と、データ処理装置の命令実行との移行をスムーズに行うことができる。

【0025】請求項5の発明に係る情報処理装置では、請求項4の情報処理装置に対して、ステージ状態管理手段は、拡張実行手段から拡張演算の実行終了通知を受けることにより、データ処理装置の命令実行ステージを終了させる。これにより、前記実行終了通知を伝達する信

号線を追加することにより、パイプライン動作を乱さず、簡単な構成で拡張演算命令への移行をスムーズに行うことができる。

【0026】請求項6の発明に係る情報処理装置では、請求項4の情報処理装置において、前記データ処理装置が、検出手段により検出された拡張演算命令に応じて、データ処理装置の実行ステージのサイクル数を決定する決定手段を備え、前記ステージ状態管理手段は、決定されたサイクル数の間命令実行ステージを継続させる。これにより、データ処理装置と拡張処理装置との間に信号線を追加することなく、また、複数の拡張処理装置の接続に好適した構成が実現できる。

【0027】請求項7の発明に係る情報処理装置では、請求項1、3または4の情報処理装置において、第1の転送命令により、拡張演算命令に必要なオペランドデータを事前に入力バッファに格納する。これにより、3つ以上のオペランドを要する拡張演算命令の実行に好適した情報処理装置を実現することができる。請求項8の発明に係る情報処理装置では、請求項7の情報処理装置において、割り込み処理の開始時に入力バッファのデータを退避バッファに退避させ、割り込み処理からの復帰時に退避バッファのデータを入力バッファに復帰させる。これにより、割り込み処理への移行を迅速にすることができる。

【0028】請求項9の発明に係る情報処理装置では、請求項7において、出力バッファは、拡張実行手段の演算結果の一部を保持する。第2の転送命令により出力バッファのデータがデータ保持手段に格納される。これにより、拡張演算の結果、複数のデータが生成される場合でも、演算結果をデータ保持手段に高速に格納することができる。

【0029】請求項10の発明に係る情報処理装置では、請求項9の情報処理装置において、割り込み処理の開始時に入力バッファと出力バッファの双方のデータを退避バッファに退避させ、割り込み処理からの復帰時に退避バッファのデータを入力バッファと出力バッファの双方に復帰させる。これにより、割り込み処理への移行を迅速にすることができ、より柔軟な拡張演算をすることができる。

【0030】請求項11の発明に係る情報処理装置では、請求項9の情報処理装置において、タスク管理手段は、複数のタスクを処理する場合に、各タスクに対応して1対の入力バッファと出力バッファとが使用される。これにより、高速なタスクスイッチを実現できる。

#### 【0031】

#### 【実施例】

<第1実施例>図1は、本発明の第1の実施例における情報処理装置の構成を示すブロック図である。この情報処理装置は、プロセッサ101、拡張演算器102、メモリ103を備え、データ処理部入力バス124、デー

タ処理部出力バス125、命令供給バス126、データバス127、データバス128、フラグデータバス130、フラグデータバス131により同図に示すように接続されている。

【0032】プロセッサ101は、デコーダ104、制御回路105、汎用レジスタ106、フラグ格納レジスタ108、命令レジスタ112、マイクロ命令レジスタ113、データラッチ114、レジスタライトバッファ116、メモリライトバッファ117、拡張演算器用データラッチ118、状態管理部129、パイプラインラッチ143、パイプラインラッチ143パイプラインラッチ143、ドライバ121~123、132~135、第1データ処理部140、第2データ処理部141を備え、命令フェッチステージ、命令解読ステージ、第1実行ステージ、第2実行ステージ、書き込みステージをからなる5段のパイプライン処理によりメモリ103の命令を実行する。メモリ103の命令にはプロセッサ101に対する命令（以後、プロセッサ命令と呼ぶ）と拡張演算器102に対する命令（以後、拡張命令と呼ぶ）とが存在するが、プロセッサ101は、命令解読ステージにおいて、拡張命令を解読した場合に、当該命令のオペランドで指定されたソースデータを汎用レジスタ106から読み出して拡張演算器102に供給し、さらに拡張演算器102による演算結果を汎用レジスタ106に格納するよう構成されている。本実施例ではプロセッサ101は32ビットプロセッサとし、特に明記していない限りバス幅も32ビットであるものとする。

【0033】拡張演算器102は、デコーダ109、制御回路110、命令レジスタ119、マイクロ命令レジスタ120、第1~第3パイプラインラッチ160~162、第1拡張処理部150、第2拡張処理部151、第3拡張処理部152、第4拡張処理部153を備え、拡張演算器102は、拡張演算として例えば乗算、除算、積和演算、ビットフィールド演算、ストリング演算、飽和演算、キューイング処理、コンテキスト切り替え処理、フィルタリング処理などを実行する複数の拡張演算器の1つであり、用途に応じて必要とされる演算種類をサポートする拡張演算器が備えられる。本実施例では、演算種類を乗算とし、命令解読ステージ、第1~第4実行ステージからなるパイプライン処理により乗算を実行するものとする。乗算を指示する拡張命令の一例を図2に、示す。同図において、「MULQDm, Dn」は、ソースレジスタDm（符号付き32ビット整数）とソースレジスタDn（符号付き32ビット整数）の内容を乗算し、結果（64ビット）の下位32ビットをディスネーションレジスタDnに、上位32ビットを汎用レジスタ106中の乗算レジスタMRに格納する。「MULQimm, Dn」は、8ビット即値データを32ビット整数に符号拡張した値とソースレジスタDn（符号付き32ビット整数）の内容を乗算し、結果をレジスタDn、MRに格



納する。「MULQ imm16, Dn」、「MULQ imm32, Dn」も同様である。

【0034】メモリ103は、プロセッサ命令及び拡張命令からなるプログラムと、プロセッサ101と拡張演算器102とで用いるデータとを記憶する。デコーダ104は、メモリ103から命令供給バス126、命令レジスタ112を介してフェッチされたプロセッサ命令及び拡張命令を命令解読ステージにおいて解読し、当該命令を実現するマイクロ命令を出力する。より詳しくいうと、デコーダ104は、解読の結果が拡張命令である場合（図2の例では、命令コードの上位4ビットがFHの場合）には、(1)拡張命令のオペランドで指定されたソースレジスタを読み出すマイクロ命令と、(2)ソースレジスタから読み出されたデータを拡張演算器用データラッチ118にラッチさせるマイクロ命令と、(3)フラグ格納レジスタ108のフラグデータをフラグデータバス130にドライバ134を介して出力させるマイクロ命令と、(4)第1、第2データ処理部140に対するNOP(no operation)を指示するマイクロ命令と、(5)拡張演算器102の演算結果をオペランドで指定されたディスティネーションレジスタに格納させるマイクロ命令と、(6)拡張演算器102からのフラグデータをドライバ135を介してフラグ格納レジスタ108に格納するマイクロ命令とを発行する。このうち前三者のマイクロ命令は、マイクロ命令レジスタ113を経由しないで直接制御回路105に出力され、他のマイクロ命令は、マイクロ命令レジスタ113を経由して制御回路105に出力される。従って、前三者のマイクロ命令は、命令解読ステージにおいて先行的に実行される。また、上記マイクロ命令は、上記(1)～(6)を内容とするものであれば良く、1つのマイクロ命令でも複数のマイクロ命令でもよい。

【0035】さらにデコーダ104は、拡張命令の種別に応じて継続数を決定し、その継続数を状態管理部129に出力する。ここで、継続数とは、プロセッサ101の実行ステージ数と拡張演算器102の実行ステージ数との差を表し、拡張命令実行中にプロセッサ101が追加すべきアイドル状態(NOP)の実行ステージ数を指す。本実施例では、拡張演算器102、プロセッサ101の実行ステージ数はそれぞれ4段、2段であるので、デコーダ104は、解読の結果、拡張命令が乗算命令である場合（図2の例では、命令コードの上位8ビットがF6H、F9H、FBHである場合）には継続数を2と決定する。

【0036】プロセッサの制御回路105は、デコーダ104から直接あるいはマイクロ命令レジスタ113を介して入力されるマイクロ命令を各種制御信号に変換するとともに、パイプライン処理を制御する回路である。拡張命令の解読結果として上記(1)、(2)、(3)のマイクロ命令が入力された場合には、当該命令の実行ステージ

に先行して当該命令の解読ステージにおいて、汎用レジスタ106の指定されたレジスタの読み出し信号及びドライバ122を開く制御信号と、拡張演算器用データラッチ118にラッチさせる制御信号と、ドライバ134を開く制御信号とを出力する。また、上記(5)、(6)のマイクロ命令が入力された場合には、汎用レジスタ106にラッチさせる制御信号及び汎用レジスタ106の指定されたレジスタの書き込み信号と、ドライバ135を開く制御信号とを出力する。

【0037】汎用レジスタ106は、32ビットデータを保持するレジスタD0～D3、乗算結果の上位32ビットを保持する乗算レジスタMRを備える。命令レジスタ112は、制御回路105の制御により、命令供給バス126を介してメモリ103からフェッチされた命令をデコーダ104へ出力する。この出力は、命令フェッチステージから命令解読ステージに変化する毎に、次の命令に切り替えられる。

【0038】フラグ格納レジスタ108は、プロセッサステータスを表す各種フラグを保持し、プロセッサ101の第2データ処理部141で生成されるフラグと、拡張演算器102の第4拡張処理部153で生成されるフラグとの双方が制御回路105の制御により反映される。マイクロ命令レジスタ113は、制御回路105の制御により、デコーダ104からのマイクロ命令を制御回路105へ出力する。この出力は、命令解読ステージから第1実行ステージに変化する毎に次のマイクロ命令（又はマイクロ命令列）に切り替えられる。

【0039】データラッチ114は、制御回路105からの制御により、データ処理部入力バス124介して汎用レジスタ106から読み出されたオペランドデータを第1データ処理部140へ出力する。この出力は、命令解読ステージから第1実行ステージに変化する毎に切り替えられる。レジスタライトバッファ116は、第2実行ステージ終了後に、制御回路105の制御により、データ処理部出力バス125からのデータを汎用レジスタ106へ出力する。

【0040】メモリライトバッファ117は、第2実行ステージ終了後に、制御回路105の制御により、データ処理部出力バス125からのデータをデータバス128上へへ出力する。拡張演算器用データラッチ118は、命令解読ステージから第1実行ステージに変化するときに、制御回路105の制御により、データ処理部入力バス124を経由するレジスタデータを拡張演算器102へ出力する。

【0041】プロセッサのドライバ121は、制御回路105の制御によりゲートを開いて、データバス127を経由するメモリ103のデータをデータ処理部入力バス124へ伝達するドライバである。ドライバ122は、制御回路105の制御によりゲートを開いて、汎用レジスタ106からデータ処理部入力バス124へデー



タを伝達するドライバである。

【0042】ドライバ123は、制御回路105の制御によりゲートを開いて、第2データ処理部141からデータ処理部出力バス125へデータを伝達するドライバである。データ処理部入力バス124は、プロセッサ101内部では汎用レジスタ106、メモリ103からのデータを第1データ処理部140及び拡張演算器用データラッチ118に伝達し、拡張演算器102内部では拡張演算器用データラッチ118からのデータを第1拡張処理部150に伝達する。注目すべき点は、データ処理部入力バス124が、プロセッサ101内部の汎用レジスタ106のデータを拡張演算器用データラッチ118を通して拡張演算器102の第1拡張処理部150に伝達する点である。このバスは、本実施例では32ビット×2本のビット数であるものとする。

【0043】データ処理部出力バス125は、プロセッサ101内の第2データ処理部141から、及び、拡張演算器102内の第1の拡張処理部153から、プロセッサ101内のレジスタライトバッファ116、メモリライトバッファ117へ実行結果を伝達する。データ処理部出力バス125は、プロセッサの第1データ処理部140と拡張演算器の第2拡張処理部153とで処理されたデータを、汎用レジスタ106メモリ103とのデータに伝達するバスである。このバスは、本実施例では32ビット×2本のビット数であるものとする。

【0044】命令供給バス126は、メモリ103に格納された命令を、プロセッサ101内の命令レジスタ112と拡張演算器102内の命令レジスタ119とに伝達するバスである。データバス127は、メモリ103に格納されたデータを、ドライバ121を介してデータ処理部入力バス124に、伝達する。

【0045】データバス128は、データ処理部出力バス125からのデータをメモリライトバッファ117を介してメモリ103へ伝達する。状態管理部129は、プロセッサ101の全てのパイプラインステージの状態を管理し、現在のステージの次のサイクルで次の処理に移るか、現在のステージをそのまま継続するかを管理する。より詳しくいうと、状態管理部129は、パイプラインステージの状態を表すフラグ情報を保持する。フラグ情報は、現在のステージの次のサイクルで次の処理に移ることを表す「次ステージ実行状態」と、次のサイクルで現在のステージをそのまま継続することを表す「現ステージ継続状態」とを表す。デコード104によりプロセッサ命令が解読された場合には、フラグ情報は「次ステージ実行状態」を示し、第1実行ステージと第2実行ステージとの2ステージ分で命令実行が終了する。デコード104により拡張命令が解読された場合には、デコード104から上記の継続数が通知され、第2次実行ステージにおいて当該継続数分のステージ数が「現ステージ継続状態」となる。例えば継続数2が通知される

と、フラグ情報は、第1実行ステージでは「次ステージ実行状態」を示し、第2実行ステージでは「現ステージ継続状態」→「現ステージ継続状態」→「次ステージ継続状態」変化する。このようにして、状態管理部129は、拡張演算器102における拡張命令の実行ステージ数に、プロセッサ101における実行ステージ数を合わせるように状態を管理する。

【0046】フラグデータバス130は、フラグ格納レジスタ108のフラグデータを拡張演算器102伝達するバスである。このバスはフラグ数と同数のビット幅でよい。フラグデータバス131は、第4拡張処理部153で処理されたフラグデータを、プロセッサ101内のフラグ格納レジスタ108に伝達するバスである。このバスもフラグ数と同数のビット幅でよい。

【0047】ドライバ132は、プロセッサの制御回路105の制御によりゲートを開いて、フラグ格納レジスタ108から第1データ処理部140へフラグデータを伝達するドライバである。ドライバ133は、制御回路105の制御によりゲートを開いて、第2データ処理部141からフラグ格納レジスタ108へフラグデータを伝達するドライバである。

【0048】ドライバ134は、制御回路105の制御によりゲートを開いて、フラグ格納レジスタ108から拡張演算器102へフラグデータを伝達するドライバである。ドライバ135は、制御回路105の制御によりゲートを開いて、拡張演算器102からフラグ格納レジスタ108へ、フラグデータを伝達するドライバである。

【0049】第1データ処理部140は、制御回路105の制御により第1実行ステージにおけるデータ処理を行う。デコード104により拡張命令が解読された場合には、制御回路105の制御によりアイドル状態(NOP)となる。第2データ処理部141は、プロセッサ101の第2実行ステージにおけるデータ処理を行う。デコード104により拡張命令が解読された場合には、制御回路105の制御によりアイドル状態(NOP)となる。

【0050】パイプラインラッチ143は、第1実行ステージから第2実行ステージに変化するときに、パイプライン制御を行うプロセッサの制御回路105からの制御により、プロセッサの第1データ処理部140から、第2データ処理部141へ、ラッチを開いてデータを伝達する。同図の拡張演算器102において、デコード109は、メモリ103から命令供給バス126、命令レジスタ112を介してフェッチされた命令を命令解読ステージにおいて解読する。解読の結果が拡張命令である場合には、拡張命令のOPコードが示す演算内容を実現するマイクロ命令を発行する。ここで注意すべきは、拡張命令のオペランドにレジスタが対して(例えば図2の乗算命令)、オペランドデータの読み出し及び書き込み

を指示するマイクロ命令を発行する必要がない点である。オペランドレジスタの読み出し及び書き込みはプロセッサ 101 により実行されるからである。

【0051】制御回路 110 は、デコーダ 109 からマイクロ命令レジスタ 120 を介して入力されるマイクロ命令を各種制御信号に変換するとともに、パイプライン処理を制御する回路である。命令レジスタ 119 は、命令供給バス 126 を介してメモリ 103 からフェッチされた命令をデコーダ 109 へ出力する。この出力は、命令フェッチステージから命令解読ステージに変化する毎に、次の命令に切り替えられる。

【0052】マイクロ命令レジスタ 120 は、命令解読ステージから第 1 実行ステージに変化するときに、パイプライン制御を行う制御回路 110 からの制御により、デコーダ 109 からのマイクロ命令を制御回路 110 へ出力する。この出力は、命令解読ステージから第 1 実行ステージに変化する毎に次のマイクロ命令（又はマイクロ命令列）に切り替えられる。

【0053】第 1 拡張処理部 150 は、拡張演算器 102 の第 1 実行ステージにおけるデータ処理を行う。第 2 拡張処理部 151 は、拡張演算器 102 の第 2 実行ステージにおけるデータ処理を行う。第 1 拡張処理部 152 は、拡張演算器 102 の第 3 実行ステージにおけるデータ処理を行う。

【0054】第 1 拡張処理部 153 は、拡張演算器 102 の第 4 実行ステージにおけるデータ処理を行う。第 1 パイプラインラッチ 160 は、第 1 実行ステージから第 2 実行ステージに変化するときに、制御回路 110 の制御により、第 1 拡張処理部 150 から第 2 拡張処理部 151 へラッチを開いてデータを伝達する。

【0055】第 2 パイプラインラッチ 161 は、第 2 実行ステージから第 3 実行ステージに変化するときに、制御回路 110 の制御により、第 2 拡張処理部 151 から第 3 拡張処理部 152 へラッチを開いてデータを伝達する。第 3 パイプラインラッチ 162 は、第 3 実行ステージから第 4 実行ステージに変化するときに、制御回路 110 からの制御により第 3 拡張処理部 152 から第 4 拡張処理部 153 へラッチを開いてデータを伝達する。

【0056】以上のように構成され本発明の第 1 の実施例における情報処理装置について、その動作を説明する。図 3 は、拡張演算器 102 において、プロセッサ 101 の汎用レジスタ 106 とフラグ格納レジスタ 108 とのデータを処理を行い、処理結果をデータを汎用レジスタ 106 とフラグ格納レジスタ 108 とに格納する動作について、パイプライン処理の状態を表すタイムチャートを示す。

【0057】同図において、“IF1” はプロセッサにおいて拡張命令をフェッチするステージである。“DP1” はプロセッサにおいて拡張命令を解読するステージである。“EXP1”、“EXP2” はそれぞれプロセッサにおいて

拡張命令を実行する（NOP）第 1、第 2 ステージである。“WB1” はプロセッサにおいて拡張演算器の演算結果を書き込むステージである。“DE1” は拡張演算器において拡張命令を命令解読するステージである。“EXE1”、“EXE2”、“EXE3”、“EXE4” は、それぞれ拡張演算器において拡張命令を実行する第 1、第 2、第 3、第 4 ステージである。“次 IF” はプロセッサにおいて IF1 でフェッチされた次の命令をフェッチするステージである。“次 DP” はプロセッサにおいて次の命令を解読するステージである。“次 EXP1”、“次 EXP2” は、それぞれプロセッサにおいて次の命令を実行する第 1、第 2 ステージである。

【0058】時刻 1 において、プロセッサの命令フェッチステージ（IF1）では、メモリ 103 は、命令をメモリからプロセッサと拡張演算器とへの命令供給バス 126 に載せる。時刻 1 から時刻 2 に変化するとき、プロセッサ 101 の制御回路 105 は、IF1 から DP1 へ移行させるため、プロセッサの命令レジスタ 112 を開ける。拡張演算器 102 の制御回路 110 は、IF1 から DE1 へ移行させるため、命令レジスタ 119 を開ける。

【0059】時刻 2 において、プロセッサの命令フェッチステージ（次 IF）では、メモリ 103 は、次命令をメモリからプロセッサと拡張演算器とへの命令供給バス 126 に載せる。プロセッサの命令解読ステージ（DP1）では、プロセッサのデコーダ 104 は、命令レジスタ 112 が開いたとき、命令を受け取り、命令を解読する。制御回路 105 は、デコーダ 104 が先行的に出力するマイクロ命令により、プロセッサのドライバ 122 を開けることにより、汎用レジスタ 106 のデータをデータ処理部入力バス 124 に載せ、ドライバ 134 を開けることによりフラグ格納レジスタ 108 のフラグ情報をフラグデータバス 130 に載せる。

【0060】拡張演算器の命令解読ステージ（DE1）では、拡張演算器のデコーダ 109 は、命令レジスタ 119 が開いたとき、命令を受け取り命令を解読する。時刻 2 から時刻 3 に変化するとき、プロセッサの制御回路 105 は、次 IF から次 DP へ移行するため、プロセッサの命令レジスタ 112 を開ける。プロセッサの制御回路 105 は、DP1 から EXP1 へパイプラインステージが変化するとき、マイクロ命令レジスタ 113、データラッチ 114、拡張演算器用データラッチ 118 を開ける。拡張演算器の制御回路 110 は、DE1 から EXE1 へパイプラインステージが変化するとき、拡張演算器のマイクロ命令レジスタ 120 を開ける。プロセッサの制御回路 105 は、プロセッサのマイクロ命令レジスタ 113 が開いたとき、プロセッサのデコーダ 104 の DP1 命令解読ステージが解読したマイクロ命令を受け取る。拡張演算器の制御回路 110 は、拡張演算器のマイクロ命令レジスタ 120 が開いたとき、拡張演算器のデコーダ 109 の命令解読ステージ（DE1）が解読したマイクロ命令を受

け取る。

【0061】時刻3において、プロセッサの命令解読ステージ (DP1) ではプロセッサのデコーダ104は、プロセッサの命令レジスタ112が開いたとき、次命令を受け取り命令を解読する。プロセッサの第1命令実行ステージ (EXP1) では、プロセッサの第1データ処理部140は、データ処理を行わない (NOP)。

【0062】拡張演算器の第1命令実行ステージ (EXE1) では、拡張演算器の第1拡張処理部150は、第1のデータ処理を行う。時刻3から時刻4に変化するとき、プロセッサの制御回路105は、次DPから次EXP1へパイプラインステージが変化するとき、プロセッサのマイクロ命令レジスタ113を開ける。プロセッサの制御回路105は、プロセッサのマイクロ命令レジスタ113が開いたとき、プロセッサのデコーダ104の次DP命令解読ステージが解読したマイクロ命令を受け取る。

【0063】プロセッサの制御回路105は、EXP1からEXP2へパイプラインステージが変化するとき、プロセッサのパイプラインラッチ143を開ける。拡張演算器の制御回路110は、EXE1からEXE2へパイプラインステージが変化するとき、拡張演算器の第1パイプラインラッチ160を開ける。時刻4において、プロセッサの第1命令実行ステージ (次EXP1) では、プロセッサの第1データ処理部140は、次実行命令の第1のデータ処理を行う。

【0064】プロセッサの第2命令実行ステージ (EXP2) では、プロセッサの第2データ処理部141は、データ処理を行わない。状態管理部129は、プロセッサの第2データ処理部141がEXP2ステージを実行することにより、状態が次ステージ実行状態から3ステージ間の現ステージ継続状態に変化し、現ステージ継続信号をプロセッサの制御回路105に伝達し続ける。

【0065】拡張演算器の第2命令実行ステージ (EXE2) では、拡張演算器の第2拡張処理部151は、データ処理を行う。時刻4から時刻5に変化するとき、プロセッサの制御回路105は、状態管理部129が伝達した現ステージ継続信号により、全てのパイプラインラッチを開かない。つまり現在のステージの状態が次のステージにおいても継続する拡張演算器の制御回路110は、EXE2からEXE3へパイプラインステージが変化するとき、拡張演算器の第2パイプラインラッチ161を開ける。

【0066】時刻5において、プロセッサの第1命令実行ステージ (次EXP1) ではステージが継続しているので、プロセッサの第1データ処理部140は、前述の次命令実行命令の第1のデータ処理を継続して行う。プロセッサの第2命令実行ステージ (EXP2) では、ステージが継続しているので、プロセッサの第2データ処理部141は、データ処理を行わない。

【0067】拡張演算器の第3命令実行ステージ (EXE

3) では、拡張演算器の第3拡張処理部152は、第3のデータ処理を行う。時刻4から時刻5に変化するとき、プロセッサの制御回路105は、状態管理部129が伝達した現ステージ継続信号により、全てのパイプラインラッチを開かない。

【0068】拡張演算器の制御回路110は、EXE3からEXE4へパイプラインステージが変化するとき、拡張演算器の第3パイプラインラッチ162を開ける。時刻6において、プロセッサの第1命令実行ステージ (次EXP1) では、ステージが継続しているので、プロセッサの第1データ処理部140は、前述の次命令実行命令の第1のデータ処理を継続して行う。

【0069】プロセッサの第2命令実行ステージ (EXP2) では、ステージが継続しているので、プロセッサの第2データ処理部141は、データ処理を行わない (NOP)。状態管理部129は、プロセッサの第2データ処理部141が3ステージ間EXP2ステージを継続した (2ステージ分が追加された) ことにより継続数2を経過したので、現ステージ継続状態から次ステージ実行状態に変化し、現ステージ継続信号をプロセッサの制御回路105に伝達を止める。

【0070】拡張演算器の第4命令実行ステージ (EXE4) では、第4拡張処理部153は、データ処理を行い、結果データとフラグ情報とを出力し、フラグデータバス131に載せる。拡張演算器の制御回路110は、拡張演算器のドライバ154を開けることにより、拡張演算器の第4拡張処理部153が処理したデータをデータ処理部出力バス125に載せる。

【0071】時刻6から時刻7に変化するとき、プロセッサの制御回路105は、次EXP1から次EXP2へパイプラインステージが変化するとき、プロセッサのパイプラインラッチ143を開ける。プロセッサの制御回路105は、EXP2からWB1へパイプラインステージが変化するとき、ドライバ123を開け、かつドライバ135を開けることによりフラグデータバス131のフラグ情報をフラグ格納レジスタ108に伝達する。

【0072】時刻7において、プロセッサの第2命令実行ステージ (次EXP2) では、プロセッサの第2データ処理部141は、次命令実行の第2のデータ処理を行う。プロセッサの書き込みステージ (WB1) では、汎用レジスタ106は、プロセッサと拡張演算器とのデータ処理部出力バス125のデータを受け取り、格納する。フラグ格納レジスタ108は、拡張演算器からプロセッサへのフラグデータバス131のフラグ情報を受け取り、格納する。

【0073】以上のように拡張演算器102は、プロセッサ101内の汎用レジスタ106のデータを直接用いて演算することができるので、メモリを介してデータの受け渡しが必要な分だけ高速に実行することができる。また、プロセッサ101内のフラグ格納レジスタ108

のフラグを参照する場合も同様に高速に実行することができる。しかも転送時のデータ不一致が生じることなく、またプロセッサ101と拡張演算器102とのパイプライン処理を乱すことなく実行時間が短縮することができる。

【0074】また、拡張演算器102が不要である場合、プロセッサ101と拡張演算器102と接続バスである拡張演算器とのデータ処理部入力バス124と、拡張演算器とのデータ処理部出力バス125と、メモリからプロセッサと拡張演算器とへの命令供給バス126と、プロセッサから拡張演算器へのフラグデータバス130と、拡張演算器からプロセッサへのフラグデータバス131とを切り離すことが可能である。

【0075】また、従来例で示すように、拡張演算器でデータを処理する場合に、処理するデータをメモリを介して拡張演算器に転送する必要がなく、実行時間を短縮することができる。さらに、実施例1では、プロセッサ101のデコーダ104が拡張命令の種類に応じて継続数を判別していたが、例えば拡張演算器102の何れかのデータ処理部から状態管理部129へ演算終了信号線を接続し、この演算終了信号により、状態管理部129のステート継続を終了するように構成してもよい。また、拡張命令自身に、プロセッサ101と拡張演算器102とのパイプライン段数の差の情報を持たせ、デコーダがこの情報より状態管理部129はステート継続回数を決定するように構成してもよい。

【0076】なお、上記実施例では、プロセッサ101に対して1つの拡張演算器102を接続した場合の構成を説明したが、種類が異なる複数の拡張演算器を接続する構成としてもよい。例えば、乗算を実行する拡張演算器と、積和演算を実行する拡張演算器と、除算を実行する拡張演算器とを接続する場合、プロセッサ101とそれぞれの拡張演算器との間で、命令供給バス126、データ処理部入力バス124、データ処理部出力バス125、フラグデータバス130、フラグデータバス131の5本のバスを接続する構成とすればよい。この場合、それぞれの拡張演算器に対する拡張命令は、その命令コードのビットパターンがユニークに割当てられている必要がある。プロセッサ101は、どの拡張演算器に対する拡張命令であっても上記実施例と同様にレジスタデータ及びフラグの供給と格納を、上記バス上に行うことになる。それぞれの拡張演算器は、命令を解釈した結果、自身に対する拡張命令のみを上記実施例と同様に実行することになる。

【0077】また、上記実施例では、プロセッサ101が2段、拡張演算器102が4段の実行ステージを持つパイプライン処理を行う例を示したが、それぞれのこれ以外の段数であってもよい。拡張演算器102がb ( $1 \leq b$ ) 段の実行ステージを持ち、プロセッサがb以下のc段 ( $1 \leq c \leq b$ ) の実行ステージを持つ場合には、第

c段目のに状態管理部を設け、b-cの値を継続数としてデコーダから状態管理部に通知するように構成すればよい。また、プロセッサの実行ステージ数cが拡張演算器の実行ステージ数bより多い場合 ( $c > b$ ) には、継続数を0として通知すればよい。この場合、プロセッサは、c段分の実行ステージが終了して時点で拡張演算器からの演算結果を取り込み、拡張演算器は、b段分の実行ステージ終了時点からプロセッサがc段分の実行ステージを終了して演算結果を取り込むまで出力することになる。

【0078】また、命令供給バス126は、プロセッサの命令レジスタ112の入力側と拡張演算器の命令レジスタ119の入力側とに接続しているが、この代わりに、命令供給バスをメモリ103からプロセッサの命令レジスタ112の入力側のみに接続し、プロセッサの命令レジスタ112の出力側から拡張演算器のデコーダ109へ接続する他のバスを設ける構成としてもよい。この場合、拡張演算器の命令レジスタ119を削除できるとともに、命令供給バスの電氣的付加を低減することができる。

【0079】さらに、本情報処理装置を1つのLSI上に構成する場合に、図12に示すレイアウト図のように、プロセッサ101については、主としてデコーダ104、制御回路105からなるコントロールモジュール部と、第1、第2データ処理部140、141とデータバス（データ処理部入力バス124、データ処理部出力バス125、命令供給バス126）等からなるデータバス部とに分類してレイアウトされる。拡張演算器102についても、主としてデコーダ109、制御回路110からなるコントロールモジュール部と、第1～第4拡張処理部150～153とデータバス等からなるデータバス部とに分類してレイアウトされる。この中で、通常、データバス部のレイアウトの一辺は、データバスのビットスライス（データバスのビット幅）に依存するので、本情報処理装置ではデータバスが共通であることから、演算種類の異なる新たな拡張演算器を容易に設計することができ、拡張演算器102を備える場合も備えない場合も容易に製造することができ、さらに、演算種別が異なる複数の拡張演算器を1チップ上に備えることも容易にできる。

<第2実施例>図4は、本発明の第2の実施例における情報処理装置の構成を示すブロック図である。この情報処理装置は、プロセッサ201、拡張演算器202、メモリ103、退避データ保持バッファ232、ドライバ259とから構成される。図1に示した第1実施例の構成と、同じ構成要素には、同じ符号を付してある。以下同じ構成要素については説明を省略し、異なる点を中心に説明する。

【0080】同図においてプロセッサ201は、説明の便宜上命令実行ステージ数は1であるものとし、命令フ

フェッチステージ、命令解読ステージ、命令実行ステージ、書き込みステージからなる4段のパイプライン処理を行うものとする。第1の実施例と異なる主要な点は、プロセッサ201と拡張演算器202との間で汎用レジスタ106のデータを転送する転送命令（以下、拡張転送命令と呼ぶ）を実行する点と、割り込み処理開始時に拡張演算器202にその旨を通知し、その通知に対する応答を確認してから割り込み処理へ分岐する点である。このため、割り込み要求信号線240、割り込み信号線241、完了信号線242が追加されている。また、データ処理部出力バス125は、第1実施例では32ビット×2構成としたが、本実施例では、32ビット×1構成でよい。

【0081】拡張演算器202は、説明の便宜上命令実行ステージ数は1であるものとし、命令フェッチステージ、命令解読ステージ、命令実行ステージ、書き込みステージからなる4段のパイプライン処理を行うものとする。第1の実施例と異なる点は、上記拡張転送命令を実行する点と、プロセッサ201から割り込み処理開始の通知を受けると、実行中の拡張命令の中断してデータを退避し、準備ができたことをプロセッサ201に回答する点である。そのため、データ処理部入力バッファ230、データ処理部出力バッファ231、退避データ保持バッファ232、その周辺にドライバ250、251、253～259が追加され、データ処理部入力バッファ退避バス243、データ処理部出力バッファ復帰バス245で接続されている。

【0082】ここで、拡張転送命令は、本実施例においてはプロセッサ201内部の汎用レジスタ106から、拡張演算器202内部のデータ処理部入力バッファ230へのデータ転送命令と、拡張演算器202内部のデータ処理部出力バッファ231から、プロセッサ201内部の汎用レジスタ106へのデータ転送命令をいう。例えば、次のように、通常の転送命令（MOVE命令）のディスチネーション、ソースにデータ処理部入力バッファ230（ex\_inBUFと略す）、データ処理部出力バッファ231（ex\_outBUFと略す）が指定された形式で表される。

MOV Dn, ex\_inBUF ; レジスタDn → ex\_inBUFへデータ転送

MOV ex\_outBUF, Dn ; ex\_outBUF → レジスタDnへデータ転送

本実施例には、拡張演算器202は、オペランドデータが3つ以上の多項演算指示する拡張命令、例えば $D_n * D_m + D_l$ という3項演算を実行するものとする。この場合、拡張転送命令は、3項演算のオペランドデータの1つDlの値を事前に拡張演算器202に転送するために利用される。また、3項演算の演算結果が複数ワードに増える場合に、演算後に演算結果の一部をプロセッサ201に転送することにも利用される。

【0083】プロセッサ201において割り込み要求信号線240は、プロセッサ201の外部からの割り込み要求をプロセッサの制御回路205へ伝達する。割り込み信号線241は、プロセッサ201の制御回路205から、拡張演算器202の制御回路210へ割り込み要求を伝達する。完了信号線242は、割り込み要求時の処理と割り込み処理完了時の処理との完了を、拡張演算器202の制御回路210からプロセッサ201の制御回路205へ伝達する。

【0084】デコーダ204は、第1実施例のデコーダ104の機能に加えて、拡張転送命令を解読する点が異なる。より具体的には、解読の結果、転送命令のソースとして汎用レジスタ106が、ディスチネーションとしてデータ処理部入力バッファ230が指定されている場合（MOV Dn, ex\_inBUFの場合）、(P1.1) レジスタDnからデータを読み出すマイクロ命令と、(P1.2) ドライバ122を開いてソースレジスタから読み出されたデータを拡張演算器用データラッチ118にラッチさせるマイクロ命令とを発行する。解読の結果、MOV ex\_outBUF, Dnの場合、(P2.1) データ処理部出力バス125からのデータをレジスタライトバッファ116にラッチさせるマイクロ命令と、(P2.2) レジスタライトバッファ116のデータをレジスタDnに書き込みマイクロ命令とを発行する。また、解読の結果、割り込み処理からの復帰を指示する命令（rti命令と略す）である場合には、(P3.1) 復帰処理（戻り先アドレスの復帰等）を指示するマイクロ命令を発行する。

【0085】制御回路205は、第1実施例の制御回路105の機能に加えて、外部から割り込み要求信号線240を介して割り込み要求を受け付けた場合に、割り込み信号線241を介して拡張演算器202の制御回路210に通知した後、拡張演算器202から完了信号線242を介して準備完了を示す応答が返ってきた時点で割り込み処理への分岐を制御する。また、上記(P1.1) (P1.2) (P2.1) (P2.2) (P3.1)の各マイクロ命令に対応する制御信号を出力する。

【0086】データ処理部207は、プロセッサ201の実行ステージにおける処理を実行する。拡張演算器202においてデコーダ209は、第1実施例のデコーダ109の機能に加えて、拡張転送命令及び割り込み復帰命令を解読する点とが異なっている。より具体的には、解読の結果、MOV Dn, ex\_inBUFである場合、(E1.1) ドライバ251を開いてデータ処理部入力バス124からのデータをデータ処理部入力バッファ230に出力させるマイクロ命令と、(E1.2) データ処理部入力バス124からドライバ251を介して伝達されるデータをデータ処理部入力バッファ230にラッチさせるマイクロ命令とを発行する。また、解読の結果、MOV ex\_outBUF, Dnの場合、(E2.1) ドライバ257を開いてデータ処理部出力バス125のデータをデータ処理部出力バス125

に出力させるマイクロ命令と、(E2.2)レジスタライトバッファ116のデータをレジスタDnに書き込みマイクロ命令とを発行する。また、解読の結果、割り込み処理からの復帰命令(rti命令と略す)である場合には、(E4.1)退避データ保持バッファ232の退避データを読み出してドライバ259を開くマイクロ命令と、(E4.2)ドライバ251を開いてデータ処理部入力バッファ230に書き込むマイクロ命令と、(E4.3)退避データ保持バッファ232の他の退避データを読み出してドライバ259を開くマイクロ命令と、(E4.4)ドライバ256を開いてデータ処理部出力バッファ231に書き込むマイクロ命令と、(E4.5)完了信号線242の出力を指示するマイクロ命令とを発行する。

【0087】制御回路210は、第1実施例の制御回路110の機能に加えて、割り込み処理開始時に退避データ保持バッファ232への退避処理を行う。より具体的には、割り込み信号線241を介して割り込み信号を受けた場合に、(E3.1)ドライバ258を開いてデータ処理部入力バッファ230のデータをデータ処理部出力バス125に出力させる制御と、(E3.2)データ処理部出力バス125上のデータを退避データ保持バッファ232に書き込む制御と、(E3.3)ドライバ257を開いてデータ処理部出力バッファ231のデータをデータ処理部出力バス125に出力する制御と、(E3.4)データ処理部出力バス125上のデータを退避データ保持バッファ232に書き足す制御と、(E3.5)完了信号線242の出力とを行う。また、上記(E1.2)(E1.2)(E2.1)(E2.2)(E4.1)(E4.2)(E4.3)(E4.5)の各マイクロ命令に対応する各種制御信号を出力する。

【0088】拡張処理部211は、拡張演算器202の実行ステージにおける拡張命令を実行する。データ処理部入力バッファ230は、拡張処理部211が複数のオペランドを用いて処理するために、プロセッサと拡張演算器とのデータ処理部入力バス124から転送されるオペランドを保持するバッファである。

【0089】データ処理部出力バッファ231は、拡張処理部211が処理し、その結果、複数のデータを出力するために、プロセッサと拡張演算器とのデータ処理部出力バス125に転送するデータを保持するバッファである。退避データ保持バッファ232は、拡張演算器のデータ処理部入力バッファ230と拡張演算器のデータ処理部出力バッファ231とのデータを退避するバッファであり、本実施例では、32ビットデータ2つ分の容量を有する先入れ先だし方式のFIFOメモリであるものとする。退避データ保持バッファ232は、プロセッサ201と拡張演算器202との外部に設置されているが、プロセッサ201が持つ、あるいは拡張演算器202が持つ場合でもよい。また、退避データ保持バッファ232は、プロセッサと拡張演算器とのデータ処理部出力バス125に接続されているが、プロセッサと拡張演

算器とのデータ処理部入力バス124とメモリからプロセッサへのデータバス127とプロセッサからメモリへのデータバス128との何れかに接続してもよい。

【0090】データ処理部入力バッファ退避バス243は、割り込み要求時に、拡張演算器のデータ処理部入力バッファ230のデータを、退避データ保持バッファ232へ退避するバスである。データ処理部入力バッファ復帰バス244は、割り込み処理完了時に、退避データ保持バッファ232に退避したデータを、拡張演算器のデータ処理部入力バッファ230へ復帰するバスである。

【0091】データ処理部出力バッファ復帰バス245は、割り込み処理完了時に、退避データ保持バッファ232に退避したデータを、拡張演算器のデータ処理部出力バッファ231へ復帰するバスである。ドライバ250は、制御回路205の制御により開いて、データ処理部入力バス124から拡張処理部211へデータを伝達する。

【0092】ドライバ251は、制御回路205の制御により開いて、データ処理部入力バス124からデータ処理部入力バッファ230へデータを伝達する。ドライバ253は、制御回路205の制御により開いて、データ処理部入力バッファ230から拡張処理部211へデータを伝達する。ドライバ254は、制御回路205の制御により開いて、拡張処理部211からデータ処理部出力バス125へデータを伝達する。

【0093】ドライバ255は、制御回路205の制御により開いて、拡張処理部211からデータ処理部出力バッファ231へデータを伝達する。ドライバ256は、制御回路205の制御により開いて、データ処理部入力バス124からデータ処理部出力バッファ231へデータを伝達する。ドライバ257は、制御回路205の制御により開いて、データ処理部出力バッファ231からデータ処理部出力バス125へデータを伝達する。

【0094】ドライバ258は、制御回路205の制御により開いて、データ処理部入力バッファ230から、プロセッサと拡張演算器とのデータ処理部出力バス125へデータを伝達する。ドライバ259は、退避データ保持バッファ232が保持している退避データを復帰するときに開いて、退避データ保持バッファ232から、プロセッサと拡張演算器とのデータ処理部入力バス124へデータを伝達する。

【0095】以上のように構成された本発明の第2の実施例における情報処理装置について、その動作を(2.1)汎用レジスタ106からデータ処理部入力バッファ230への拡張転送命令の動作、(2.2)データ処理部入力バッファ230のオペランドデータを用いる拡張命令の動作、(2.3)データ処理部出力バッファ231のデータから汎用レジスタ106への拡張転送命令の動作、(2.4)割り込み処理開始時のデータ処理部入

力バッファ 230 およびデータ処理部出力バッファ 231 のデータ退避動作、(2. 5) 割り込み処理終了時のデータ処理部入力バッファ 230 及びデータ処理部出力バッファ 231 へのデータ復帰動作に分けて説明する。

【0096】(2. 1) データ処理部入力バッファ 230 への書き込み動作

図5は、プロセッサの汎用レジスタ 106 が保持するデータをデータ処理部入力バッファ 230 へ書き込む拡張転送命令の実行時のパイプライン処理の状態を表すタイムチャートである。同図において、“IF1”はプロセッサにおいて拡張転送命令をフェッチするステージである。“DPI”はプロセッサにおいて拡張転送命令を解読するステージである。“EXPI”はプロセッサにおいて拡張転送命令を命令実行する第1ステージでありデータ処理は行わない(NOP)。“WB1”はプロセッサにおいて拡張転送命令を演算結果を書き込むステージである。“DE1”は拡張演算器において、拡張転送命令を解読するステージである。“EXE1”は拡張演算器において拡張演算器の実行命令を実行するステージである。“入力保持オペランド”はデータ処理部入力バッファ 230 が書き込まれたデータを保持し定理タイミングを表す。

【0097】時刻1において、プロセッサの命令フェッチステージでは、メモリ 103 は、命令をメモリからプロセッサと拡張演算器とへの命令供給バス 126 に載せる。時刻1から時刻2に変化するとき、プロセッサの制御回路 205 は、IF1 から DPI へ移行するため、プロセッサの命令レジスタ 112 を開ける。拡張演算器の制御回路 210 は、IF1 から DE1 へ移行するため、拡張演算器の命令レジスタ 119 を開ける。

【0098】時刻2において、プロセッサの命令解読ステージ(DPI)では、プロセッサのデコーダ 204 は、プロセッサの命令レジスタ 112 が開いたとき、命令を受け取り解読する。プロセッサの制御回路 205 は、プロセッサのデコーダ 204 が先行的に出力するマイクロ命令により、プロセッサのドライバ 122 を開けることにより、プロセッサの汎用レジスタ 106 のオペランドをデータ処理部入力バス 124 に載せる。

【0099】拡張演算器の命令解読ステージ(DE1)では、拡張演算器のデコーダ 209 は、拡張演算器の命令レジスタ 119 が開いたとき、命令を受け取り、命令を解読する。時刻2から時刻3に変化するとき、プロセッサの制御回路 205 は、DPI から EXPI へパイプラインステージが変化するとき、プロセッサのマイクロ命令レジスタ 113、114、218 を開け、その結果、プロセッサのデコーダ 204 から、プロセッサの制御回路 205 へ、ラッチを開いてマイクロ命令を伝達し、パイプライン制御を行うプロセッサの制御回路 205 からの制御により、プロセッサと拡張演算器とのデータ処理部入力バス 124 から、プロセッサのデータ処理部 207 へ、ラッチを開いてオペランドを伝達し、プロセッサと

拡張演算器とのデータ処理部入力バス 124 のデータが、拡張演算器 202 に達する。拡張演算器の制御回路 210 は、DE1 から EXE1 へパイプラインステージが変化するとき、拡張演算器のマイクロ命令レジスタ 120 を開ける。プロセッサの制御回路 205 は、プロセッサのマイクロ命令レジスタ 113 が開いたとき、プロセッサのデコーダ 204 の DPI 命令解読ステージが解読したマイクロ命令を受け取る。拡張演算器の制御回路 210 は、拡張演算器のマイクロ命令レジスタ 120 が開いたとき、拡張演算器のデコーダ 209 の DE1 命令解読ステージが解読したマイクロ命令を受け取る。

【0100】時刻3において、プロセッサの命令実行ステージ(EXPI)では、プロセッサのデータ処理部 207 は、データ処理を行わない。拡張演算器の命令実行ステージ(EXE1)では、拡張演算器の制御回路 210 は、拡張演算器のドライバ 251 を開けることにより、プロセッサと拡張演算器とのデータ処理部入力バス 124 の1個(32ビット)のオペランドを拡張演算器のデータ処理部入力バッファ 230 に伝達する。拡張演算器のデータ処理部入力バッファ 230 は、プロセッサと拡張演算器とのデータ処理部入力バス 124 の1個のオペランドを保持する。

【0101】(2. 2) データ処理部入力バッファ 230 のオペランドデータを用いる拡張演算命令の実行動作図6は、拡張演算器 202 において、プロセッサの汎用レジスタ 106 が保持するデータとデータ処理部入力バッファ 230 が保持するデータとを用いて、3項演算を行って、演算結果の一部分をプロセッサの汎用レジスタ 106 に転送し、演算結果の他の部分をデータ処理部出力バッファ 231 に書き込む拡張命令の動作について、パイプライン処理の状態を表すタイムチャートである。同図中“出力保持データ”はデータ処理部出力バッファ 231 にデータが保持されているタイミングを表す。

【0102】時刻1において、プロセッサの命令フェッチステージ(IF1)では、メモリ 103 は、命令をメモリからプロセッサと拡張演算器とへの命令供給バス 126 に載せる。時刻1から時刻2に変化するとき、プロセッサの制御回路 205 は、IF1 から DPI へプロセッサの命令レジスタ 112 を開ける。拡張演算器の制御回路 210 は、IF1 から DE1 へ移行するため、拡張演算器の命令レジスタ 119 を開ける。

【0103】時刻2において、プロセッサの命令解読ステージ(DPI)では、プロセッサのデコーダ 204 は、プロセッサの命令レジスタ 112 が開いたとき、命令を受け取り、命令を解読する。プロセッサの制御回路 205 は、プロセッサのデコーダ 204 が先行的に出力するマイクロ命令により、プロセッサのドライバ 122 を開けることにより、プロセッサの汎用レジスタ 106 のオペランドをプロセッサと拡張演算器とのデータ処理部入力バス 124 に載せる。



【0104】拡張演算器の命令解読ステージ(DEI)では、拡張演算器のデコーダ209は、拡張演算器の命令レジスタ119が開いたとき、命令を受け取り、命令を解読する。時刻2から時刻3に変化するとき、プロセッサの制御回路205は、DPIからEXPIへパイプラインステージが変化するとき、プロセッサのマイクロ命令レジスタ113、114、218を開け、その結果、プロセッサのデコーダ204からプロセッサの制御回路205へ、ラッチを開いてマイクロ命令を伝達し、パイプライン制御を行うプロセッサの制御回路205からの制御により、プロセッサと拡張演算器とのデータ処理部入力バス124から、プロセッサのデータ処理部207へ、ラッチを開いてオペランドを伝達し、プロセッサと拡張演算器とのデータ処理部入力バス124のデータが、拡張演算器202に達する。拡張演算器の制御回路210は、DEIからEXEIへパイプラインステージが変化するとき、拡張演算器のマイクロ命令レジスタ120を開ける。プロセッサの制御回路205は、プロセッサのマイクロ命令レジスタ113が開いたとき、プロセッサのデコーダ204のDPI命令解読ステージが解読したマイクロ命令を受け取る。拡張演算器の制御回路210は、拡張演算器のマイクロ命令レジスタ120が開いたとき、拡張演算器のデコーダ209のDEI命令解読ステージが解読したマイクロ命令を受け取る。

【0105】時刻3において、プロセッサの命令実行ステージ(EXPI)では、プロセッサのデータ処理部207は、データ処理を行わない。拡張演算器の命令実行ステージ(EXEI)では、拡張演算器の制御回路210は、拡張演算器のドライバ250を開けることにより、プロセッサと拡張演算器とのデータ処理部入力バス124の1個のオペランドを拡張処理部211に伝達し、拡張演算器のドライバ253を開けることにより、拡張演算器のデータ処理部入力バッファ230の1個のオペランドを拡張処理部211に伝達する。拡張処理部211は、プロセッサと拡張演算器とのデータ処理部入力バス124の1個のオペランドと、拡張演算器のデータ処理部入力バッファ230の1個のオペランドとを処理し結果として2個の処理結果データを出力する。拡張演算器の制御回路210は、拡張演算器のドライバ254を開けることにより、拡張処理部211の一方の処理結果データをプロセッサと拡張演算器とのデータ処理部出力バス125に伝達し、プロセッサと拡張演算器とのデータ処理部出力バス125に載せることにより、処理結果データがプロセッサ201に達する。拡張演算器の制御回路210は、拡張演算器のドライバ255を開けることにより、拡張処理部211のもう一方の処理結果データを拡張演算器のデータ処理部出力バッファ231に伝達する。拡張演算器のデータ処理部出力バッファ231は、もう一方の処理結果データを保持する。

【0106】時刻2から時刻3に変化するとき、プロセ

ッサの制御回路205は、EXPIからWBIへパイプラインステージが変化するとき、プロセッサのレジスタライトバッファ116を開け、その結果、プロセッサと拡張演算器とのデータ処理部出力バス125から、プロセッサの汎用レジスタ106へ、ラッチを開いて一方の処理結果データを伝達する。

【0107】時刻4において、プロセッサの汎用レジスタ106は、一方の処理結果データを保持する。

(2.3)データ処理部出力バッファ231のデータを読み出して汎用レジスタ106に格納する動作

図7は、データ処理部出力バッファ231が保持するデータをプロセッサの汎用レジスタ106へ書き込む拡張転送命令の動作について、パイプライン処理の状態を表すタイムチャートである。

【0108】時刻1において、プロセッサの命令フェッチステージでは、メモリ103は、命令をメモリからプロセッサと拡張演算器との命令供給バス126に載せる。時刻1から時刻2に変化するとき、プロセッサの制御回路205は、IFIからDPIへ移行するため、プロセッサの命令レジスタ112を開ける。拡張演算器の制御回路210は、IFIからDEIへ移行するため、拡張演算器の命令レジスタ119を開ける。

【0109】時刻2において、プロセッサの命令解読ステージ(DPI)では、プロセッサのデコーダ204は、プロセッサの命令レジスタ112が開いたとき、命令を受け取り、命令を解読する。拡張演算器の命令解読ステージ(DEI)では、拡張演算器のデコーダ209は、拡張演算器の命令レジスタ119が開いたとき、命令を受け取り、命令を解読する。

【0110】時刻2から時刻3に変化するとき、プロセッサの制御回路205は、DPIからEXPIへパイプラインステージが変化するとき、プロセッサのマイクロ命令レジスタ113、114を開け、その結果、プロセッサのデコーダ204から、プロセッサの制御回路205へ、ラッチを開いてマイクロ命令を伝達し、パイプライン制御を行うプロセッサの制御回路205からの制御により、プロセッサと拡張演算器とのデータ処理部入力バス124から、プロセッサのデータ処理部207へ、ラッチを開いてオペランドを伝達する。拡張演算器の制御回路210は、DEIからEXEIへパイプラインステージが変化するとき、拡張演算器のマイクロ命令レジスタ120を開ける。プロセッサの制御回路205は、プロセッサのマイクロ命令レジスタ113が開いたとき、プロセッサのデコーダ204のDPI命令解読ステージが解読したマイクロ命令を受け取る。拡張演算器の制御回路210は、拡張演算器のマイクロ命令レジスタ120が開いたとき、拡張演算器のデコーダ209のDEI命令解読ステージが解読したマイクロ命令を受け取る。

【0111】時刻3において、プロセッサの命令実行ステージ(EXPI)では、プロセッサのデータ処理部207

は、データ処理を行わない。拡張演算器の命令実行ステージ (EXE1) では、拡張演算器の制御回路 210 は、拡張演算器のドライバ 257 を開けることにより、拡張演算器のデータ処理部出力バッファ 231 が保持する処理結果データを、プロセッサと拡張演算器とのデータ処理部出力バス 125 に伝達する。

【0112】時刻 2 から時刻 3 に変化するとき、プロセッサの制御回路 205 は、EXPI から WBI へパイプラインステージが変化するとき、プロセッサのレジスタライトバッファ 116 を開け、その結果、プロセッサと拡張演算器とのデータ処理部出力バス 125 から、プロセッサの汎用レジスタ 106 へ、ラッチを開いて一方の処理結果データを伝達する。

【0113】時刻 4 において、プロセッサの汎用レジスタ 106 は、一方の処理結果データを保持する。このように拡張演算器がバッファを持つことにより、チップ面積の増大によるコスト上昇と消費電力の増大することなく拡張演算器が複数のオペランドを処理することができる。

【0114】(2. 4) 割り込み処理開始時のデータ処理部入力バッファ 230 およびデータ処理部出力バッファ 231 のデータ退避動作。

図 8 は、プロセッサ 201 に割り込みが発生し、拡張演算器のデータ処理部入力バッファ 230 と、拡張演算器のデータ処理部出力バッファ 231 とのデータを、退避データ保持バッファ 232 に格納する動作を表すタイムチャートである。このタイムチャートでは、“ESC1” は、拡張演算器において拡張演算器のデータ処理部入力バッファ 230 を退避する命令を行うことを表す。“ESC2” は、拡張演算器において拡張演算器のデータ処理部出力バッファ 231 を退避する命令を行うことを表す。

【0115】時刻 1 において、プロセッサ 201 は、外部からの割り込み要求信号線 240 より割り込み要求信号を受け取る。時刻 2 において、プロセッサの制御回路 205 は、プロセッサから拡張演算部への割り込み信号線 241 に割り込み信号を伝達する。拡張演算器の制御回路 210 は、プロセッサから拡張演算部への割り込み信号線 241 より割り込み信号を受け取る。

【0116】時刻 3 において、拡張演算器の制御回路 210 は、拡張演算器のドライバ 258 を開けることにより、拡張演算器のデータ処理部入力バッファ 230 の処理結果をプロセッサと拡張演算器とのデータ処理部出力バス 125 に伝達し、プロセッサと拡張演算器とのデータ処理部出力バス 125 に載せる。退避データ保持バッファ 232 は、プロセッサと拡張演算器とのデータ処理部出力バス 125 よりデータを受け取り格納を更新する。

【0117】時刻 4 において、拡張演算器の制御回路 210 は、拡張演算器のドライバ 257 を開けることにより、拡張演算器のデータ処理部出力バッファ 231 の処

理結果をプロセッサと拡張演算器とのデータ処理部出力バス 125 に伝達し、プロセッサと拡張演算器とのデータ処理部出力バス 125 に載せる。退避データ保持バッファ 232 は、プロセッサと拡張演算器とのデータ処理部出力バス 125 よりデータを受け取り、格納する。

【0118】時刻 5 において、拡張演算器の制御回路 210 は、拡張演算部からプロセッサへの割り込み手続き完了信号線 242 に割り込み準備完了信号を伝達し、プロセッサ 201 に割り込み処理の許可を与える。

(2. 5) 割り込み処理終了時のデータ処理部入力バッファ 230 及びデータ処理部出力バッファ 231 へのデータ復帰動作。

【0119】図 9 は、割り込み復帰命令のパイプライン処理の状態を表すタイムチャートである。このタイムチャートでは、“RT11” は、プロセッサ 201 及び拡張演算器 202 においてデータ処理部入力バッファ 230 へのデータ復帰処理を表す。“RT12” は、プロセッサ 201 及び拡張演算器 202 において、データ処理部出力バッファ 231 へのデータ復帰処理を表す。

【0120】次いで、拡張演算器 202 において、割り込み復帰命令の動作について図 9 のパイプライン処理の状態を表すタイムチャートを用いて説明する。時刻 1 において、プロセッサの命令フェッチステージでは、メモリ 103 は、割り込み復帰命令をメモリからプロセッサと拡張演算器とへの命令供給バス 126 に載せる。

【0121】時刻 1 から時刻 2 に変化するとき、プロセッサの制御回路 205 は、プロセッサの命令レジスタ 112 を開ける。拡張演算器の制御回路 210 は、IF1 から DE1 へ移行するため、拡張演算器の命令レジスタ 119 を開ける。時刻 2 において、プロセッサの命令解読ステージ (RT11) では、プロセッサのデコーダ 204 は、プロセッサの命令レジスタ 112 が開いたとき、割り込み復帰命令を受け取り、割り込み復帰命令を解読する。

【0122】拡張演算器の命令解読ステージ (RT11) では、拡張演算器のデコーダ 209 は、拡張演算器の命令レジスタ 119 が開いたとき、割り込み復帰命令を受け取り、割り込み復帰命令を解読する。時刻 2 から時刻 3 に変化するとき、プロセッサの制御回路 205 は、RT11 処理で、パイプラインステージが変化するとき、プロセッサのマイクロ命令レジスタ 113、114 を開け、その結果、プロセッサのデコーダ 204 から、プロセッサの制御回路 205 へ、ラッチを開いて割り込み復帰マイクロ命令を伝達する。拡張演算器の制御回路 210 は、RT11 処理で、パイプラインステージが変化するとき、拡張演算器のマイクロ命令レジスタ 120 を開ける。プロセッサの制御回路 205 は、プロセッサのマイクロ命令レジスタ 113 が開いたとき、プロセッサのデコーダ 204 の命令解読ステージが RT11 を解読したマイクロ命令を受け取る。拡張演算器の制御回路 210 は、拡張演算器のマイクロ命令レジスタ 120 が開いたと

き、拡張演算器のデコーダ 209 の命令解読ステージが RT11 を解読したマイクロ命令を受け取る。

【0123】時刻 3 において、プロセッサの命令解読ステージ (RT12) では、割り込み復帰マイクロ命令を繰り返し出力する。プロセッサの命令実行ステージ (RT11) では、プロセッサの復帰処理を行う。拡張演算器の命令解読ステージ (RT12) では、割り込み復帰マイクロ命令を繰り返し出力する。

【0124】拡張演算器の命令実行ステージ (RT11) では、退避データ保持バッファ 232 は、退避データ保持バッファのドライバ 259 を開いて、プロセッサと拡張演算器とのデータ処理部入力バス 124 にデータを載せる。拡張演算器の制御回路 210 は、拡張演算器のドライバ 256 を開けることにより、プロセッサと拡張演算器とのデータ処理部入力バス 124 のデータを拡張演算器のデータ処理部入力バッファ 230 に伝達する。拡張演算器のデータ処理部入力バッファ 230 は、プロセッサと拡張演算器とのデータ処理部入力バス 124 のデータを受け取り、保持する。

【0125】時刻 3 において、プロセッサの命令解読ステージ (RT13) では、割り込み復帰マイクロ命令を繰り返し出力する。プロセッサの命令実行ステージ (RT12) では、プロセッサの復帰処理を行う。拡張演算器の命令実行ステージ (RT12) では、退避データ保持バッファ 232 は、退避データ保持バッファのドライバ 259 を開いて、プロセッサと拡張演算器とのデータ処理部入力バス 124 にデータを載せる。拡張演算器の制御回路 210 は、拡張演算器のドライバ 256 を開けることにより、プロセッサと拡張演算器とのデータ処理部入力バス 124 のデータを拡張演算器のデータ処理部出力バッファ 231 に伝達する。拡張演算器のデータ処理部出力バッファ 231 は、プロセッサと拡張演算器とのデータ処理部入力バス 124 のデータを受け取り、格納する。

<第 3 実施例> 図 10 は、本発明の実施例 3 における情報処理装置の構成を示すブロック図である。

【0126】この情報処理装置は、プロセッサ 301 と、拡張演算器 302 と、メモリ 103 とから構成される。図 4 に示した第 2 実施例の構成と、同じ構成要素には、同じ符号を付してある。退避データ保持バッファについては、図示と説明を省略する。以下、同じ構成要素については説明を省略し、異なる点を中心に説明する。同図においてプロセッサ 301 は、第 2 に実施例のプロセッサ 201 の機能に加えて、タスク切り替え機能を有している。プロセッサ 301 のタスクの切り替えについては、本実施例ではタスク切り替え命令によって切り替え処理を実行するものとする。ここで、タスク切り替え命令は、切り替え先のタスク番号の指定するとともにタスク切り替えを指示する命令である。プロセッサ 301 のタスク切り替えは一般的な方法でよい。また、データ

処理部出力バス 125 は、第 2 実施例と同様に 32 ビット×1 構成でよい。

【0127】拡張演算器 302 は、第 2 の実施例の拡張演算器 202 のデータ処理部入力バッファ 230、データ処理部出力バッファ 231、及び、その周辺のドライバの代わりに、データ処理部入力バッファ 330、331、データ処理部出力バッファ 332、333、その周辺のドライバを備え、タスク管理部 361 が新たに追加されている。

【0128】プロセッサ 301 においてデコーダ 304 は、第 2 実施例のデコーダ 204 の機能に加えて、タスク切り替え命令を解読し、当該命令で指定されたタスクに切り替えるマイクロ命令を発行する点異なる。プロセッサの制御回路 305 は、第 2 実施例の制御回路 205 の機能に加えて、タスク切り替えを実現するマイクロ命令を受けると、それに対応する各種制御信号を出力する。

【0129】拡張演算器 302 においてデコーダ 309 は、第 2 実施例のデコーダ 209 の機能に加えて、タスク切り替え命令を解読する点異なる。具体的には、解読結果がタスク切り替え命令である場合には、当該命令で指定されたタスク番号をタスク管理部 361 に格納するマイクロ命令を発行する。プロセッサのデータ処理部 207 は、プロセッサ 301 の主なデータ処理を行う。

【0130】制御回路 310 は、第 2 の実施例の制御回路 210 の機能に加えて、タスク切り替え命令のデコード結果としてのマイクロ命令を受け付けると、当該命令で指定されたタスク番号をタスク管理部 361 に格納する制御信号を出力する。また、第 2 実施例で説明した拡張転送命令のデコード結果としてのマイクロ命令を受け付けると、タスク管理部 361 のタスク番号に応じてデータ処理部入力バッファ 330 とデータ処理部入力バッファ 331 の一方を選択的に使用する。例えば、タスク管理部 361 のタスク番号が 1 のときには、データ処理部入力バッファ 330、データ処理部出力バッファ 332 を使用し、タスク番号が 2 のときには、データ処理部入力バッファ 331、データ処理部出力バッファ 333 を使用するよう制御する。また例えば、3 項演算を行う拡張命令の場合には、データ処理部入力バッファ 330 と 331 とを選択的に使用し、演算結果が複数のデータになる拡張命令の場合には、タスク番号に応じてデータ処理部出力バッファ 332 と 333 とを選択的に使用するよう制御する。

【0131】データ処理部入力バッファ 330、331 は、それぞれ拡張処理部 211 が複数のオペランドを用いて処理する場合に、データ処理部入力バス 124 から転送されるデータを保持するバッファであり、タスク管理部 361 のタスク番号に応じて選択される。データ処理部出力バッファ 332、333 は、それぞれ拡張処理部 211 の演算結果が複数のデータになる場合に、デー

タ処理部出力バス 125 に転送すべき当該データを保持するバッファであり、タスク管理部 361 のタスク番号に応じて選択される。

【0132】ドライバ 350 は、制御回路 305 の制御によりドライバを開いて、プロセッサと拡張演算器とのデータ処理部入力バス 124 から拡張処理部 211 へデータを伝達する。ドライバ 351 は、制御回路 305 の制御によりドライバを開いて、プロセッサと拡張演算器とのデータ処理部入力バス 124 からデータ処理部入力バッファ 330 へデータを伝達する。

【0133】ドライバ 352 は、制御回路 305 の制御によりドライバを開いて、プロセッサと拡張演算器とのデータ処理部入力バス 124 からデータ処理部入力バッファ 331 へデータを伝達する。ドライバ 354 は、制御回路 305 の制御により、拡張処理部 211 から、プロセッサと拡張演算器とのデータ処理部出力バス 125 へ、ドライバを開いてデータを伝達する。

【0134】ドライバ 355 は、制御回路 305 の制御によりドライバを開いて、拡張処理部 211 からデータ処理部出力バッファ 332 へデータを伝達する。ドライバ 356 は、制御回路 305 の制御によりドライバを開いて、拡張処理部 211 からデータ処理部出力バッファ 333 へデータを伝達する。ドライバ 357 は、制御回路 305 の制御によりドライバを開いて、データ処理部出力バッファ 332 からプロセッサと拡張演算器とのデータ処理部出力バス 125 へデータを伝達する。

【0135】ドライバ 358 は、制御回路 305 の制御によりドライバを開いて、データ処理部出力バッファ 333 からプロセッサと拡張演算器とのデータ処理部出力バス 125 へデータを伝達する。セレクト 360 は、制御回路 305 の制御により、データ処理部入力バッファ 330 とデータ処理部入力バッファ 331 とを選択する。

【0136】タスク管理部 361 は、制御回路 305 の制御により、タスク番号を保持し、タスク番号を制御回路 305 に伝達する。本実施例では、タスク管理部 361 が保持するタスク番号は、タスク 1、タスク 2 とし、タスク管理部 361 がタスク 1 を保持する場合、データ処理部入力バッファ 330 と、データ処理部出力バッファ 332 とが選択され、タスク管理部 361 がタスク 2 を保持する場合、データ処理部入力バッファ 331 と、データ処理部出力バッファ 333 とが選択されるものとする。

【0137】以上のように構成された本発明の第 3 の実施例における情報処理装置について、その動作を (3.

1) タスク切り替え命令の実行動作、(3. 2) プロセッサの汎用レジスタ 106 が保持するデータを拡張演算器 302 のデータ処理部入力バッファへ書き込む拡張転送命令の動作、(3. 3) プロセッサの汎用レジスタ 106 のデータと、データ処理部入力バッファのデータと

を用いて演算し、出力結果をプロセッサの汎用レジスタ 106 とデータ処理部出力バッファとへの書き込みを行う拡張命令の動作、(3. 4) プロセッサ 201 のデータ処理部出力バッファのデータをプロセッサの汎用レジスタ 106 へ書き込む拡張転送命令の動作に分けて説明する。

【0138】(3. 1) 拡張演算器 302 におけるタスク切り替える命令の実行動作。

命令フェッチステージにおいて、メモリ 103 は、タスク切り替え命令を命令供給バス 126 に載せる。命令解読ステージにおいて、制御回路 310 は、命令レジスタ 119 を開ける。デコーダ 309 は、命令レジスタ 119 が開いたとき、命令を受け取り、タスク切り替え命令を解読する。拡張演算器の制御回路 310 は、パイプラインステージが変化するとき、マイクロ命令レジスタ 120 を開ける。制御回路 310 は、拡張演算器のマイクロ命令レジスタ 120 が開いたとき、拡張演算器のデコーダ 309 が解読したマイクロ命令を受け取り、タスク切り換えを行う制御を行うとともに、タスク管理部 361 に命令で指定されたタスク番号を格納する。拡張転送命令および 3 項以上の演算を行う拡張命令の実行時に、このタスク番号がタスク 1 であれば、データ処理部入力バッファ 330 とデータ処理部出力バッファ 332 とが選択され、タスク番号がタスク 2 であれば、データ処理部入力バッファ 331 とデータ処理部出力バッファ 333 とが選択される。

【0139】(3. 2) 拡張演算器 302 において、タスク管理部 361 がタスク 1 を保持し、プロセッサの汎用レジスタ 106 が保持するデータをデータ処理部入力バッファ 330 へ書き込む命令の動作について説明する。図 5 に示したパイプライン処理の状態を表すタイムチャートは、本動作においても同じタイミングであるので、図 5 を参照しつつ説明する。

【0140】時刻 1 において、プロセッサの命令フェッチステージでは、メモリ 103 は、命令をメモリからプロセッサと拡張演算器とへの命令供給バス 126 に載せる。時刻 1 から時刻 2 に変化するとき、プロセッサの制御回路 305 は、IF1 から DPI へ移行するため、プロセッサの命令レジスタ 112 を開ける。拡張演算器の制御回路 310 は、IF1 から DE1 へ移行するため、拡張演算器の命令レジスタ 119 を開ける。

【0141】時刻 2 において、プロセッサの命令解読ステージ (DPI) では、プロセッサのデコーダ 304 は、プロセッサの命令レジスタ 112 が開いたとき、命令を受け取り、命令を解読する。プロセッサの制御回路 305 は、プロセッサのデコーダ 304 が先行的に出力するマイクロ命令により、プロセッサのドライバ 122 を開けることにより、プロセッサの汎用レジスタ 106 のオペランドをプロセッサと拡張演算器とのデータ処理部入力バス 124 に載せる。

【0142】拡張演算器の命令解読ステージ(DEI)では、拡張演算器のデコーダ309は、拡張演算器の命令レジスタ119が開いたとき、命令を受け取り、命令を解読する。時刻2から時刻3に変化するとき、プロセッサの制御回路305は、DPIからEXPIへパイプラインステージが変化するとき、プロセッサのマイクロ命令レジスタ113、314、318を開け、その結果、プロセッサのデコーダ304から、プロセッサの制御回路305へ、ラッチを開いてマイクロ命令を伝達し、パイプライン制御を行うプロセッサの制御回路305の制御により、プロセッサと拡張演算器とのデータ処理部入力バス124から、プロセッサのデータ処理部207へ、ラッチを開いてオペランドを伝達し、プロセッサと拡張演算器とのデータ処理部入力バス124のデータが、拡張演算器302に達する。拡張演算器の制御回路310は、DEIからEXEIへパイプラインステージが変化するとき、拡張演算器のマイクロ命令レジスタ120を開ける。プロセッサの制御回路305は、プロセッサのマイクロ命令レジスタ113が開いたとき、プロセッサのデコーダ304のDPI命令解読ステージが解読したマイクロ命令を受け取る。拡張演算器の制御回路310は、拡張演算器のマイクロ命令レジスタ120が開いたとき、拡張演算器のデコーダ309のDEI命令解読ステージが解読したマイクロ命令を受け取る。

【0143】時刻3において、プロセッサの命令実行ステージ(EXPI)では、プロセッサのデータ処理部207は、データ処理を行わない。拡張演算器の命令実行ステージ(EXEI)では、タスク管理部361は、保持するタスク情報としてタスク1を、拡張演算器の制御回路310へ出力する。拡張演算器の制御回路310は、タスク管理部361が出力するタスク1により、拡張演算器のドライバ351を開けることにより、プロセッサと拡張演算器とのデータ処理部入力バス124の1個のオペランドを拡張演算器のデータ処理部入力バッファ330に伝達する。拡張演算器のデータ処理部入力バッファ330は、プロセッサと拡張演算器とのデータ処理部入力バス124の1個のオペランドを保持する。

【0144】(3.3)拡張演算器302においてタスク管理部361がタスク1を保持している場合に、プロセッサの汎用レジスタ106が保持するデータと、データ処理部入力バッファ330が保持するデータとを用いて演算し、出力結果をプロセッサの汎用レジスタ106と、データ処理部出力バッファ332とへの書き込みを行う拡張命令の動作について説明する。図6に示したパイプライン処理の状態を表すタイムチャートは、本動作においても同じタイミングであるので、図6を参照しつつ説明する。

【0145】時刻1において、プロセッサの命令フェッチステージでは、メモリ103は、命令をメモリからプロセッサと拡張演算器とへの命令供給バス126に載せ

る。時刻1から時刻2に変化するとき、プロセッサの制御回路305は、IFIからDPIへ移行するため、プロセッサの命令レジスタ112を開ける。拡張演算器の制御回路310は、IFIからDEIへ移行するため、拡張演算器の命令レジスタ119を開ける。

【0146】時刻2において、プロセッサの命令解読ステージ(DPI)では、プロセッサのデコーダ304は、プロセッサの命令レジスタ112が開いたとき、命令を受け取り、命令を解読する。プロセッサの制御回路305は、プロセッサのデコーダ304が先行的に出力するマイクロ命令により、プロセッサのドライバ122を開けることにより、プロセッサの汎用レジスタ106のオペランドをプロセッサと拡張演算器とのデータ処理部入力バス124に載せる。

【0147】拡張演算器の命令解読ステージ(DEI)では、拡張演算器のデコーダ309は、拡張演算器の命令レジスタ119が開いたとき、命令を受け取り、命令を解読する。時刻2から時刻3に変化するとき、プロセッサの制御回路305は、DPIからEXPIへパイプラインステージが変化するとき、プロセッサのマイクロ命令レジスタ113、314、318を開け、その結果、プロセッサのデコーダ304から、プロセッサの制御回路305へ、ラッチを開いてマイクロ命令を伝達し、パイプライン制御を行うプロセッサの制御回路305の制御により、プロセッサと拡張演算器とのデータ処理部入力バス124から、プロセッサのデータ処理部207へ、ラッチを開いてオペランドを伝達し、プロセッサと拡張演算器とのデータ処理部入力バス124のデータが、拡張演算器302に達する。拡張演算器の制御回路310は、DEIからEXEIへパイプラインステージが変化するとき、拡張演算器のマイクロ命令レジスタ120を開ける。プロセッサの制御回路305は、プロセッサのマイクロ命令レジスタ113が開いたとき、プロセッサのデコーダ304のDPI命令解読ステージが解読したマイクロ命令を受け取る。拡張演算器の制御回路310は、拡張演算器のマイクロ命令レジスタ120が開いたとき、拡張演算器のデコーダ309のDEI命令解読ステージが解読したマイクロ命令を受け取る。

【0148】時刻3において、プロセッサの命令実行ステージ(EXPI)では、プロセッサのデータ処理部207は、データ処理を行わない。拡張演算器の命令実行ステージ(EXEI)では、タスク管理部361は、保持するタスク情報としてタスク1を、拡張演算器の制御回路310へ出力する。拡張演算器の制御回路310は、拡張演算器のドライバ350を開けることにより、プロセッサと拡張演算器とのデータ処理部入力バス124の1個のオペランドを拡張演算器のデータ処理部211に伝達し、タスク管理部361が出力するタスク1により、拡張演算器のセクタ360が拡張演算器のデータ処理部入力バッファ330を選択し、1個のオペランドを拡張演算器のデータ処理部211に伝

達する。拡張処理部211は、プロセッサと拡張演算器とのデータ処理部入力バス124の1個のオペランドと、拡張演算器のデータ処理部入力バッファ330の1個のオペランドとを処理し結果として2個の処理結果データを出力する。拡張演算器の制御回路310は、拡張演算器のドライバ354を開けることにより、拡張処理部211の一方の処理結果データをプロセッサと拡張演算器とのデータ処理部出力バス125に伝達し、プロセッサと拡張演算器とのデータ処理部出力バス125に載せることにより、処理結果データがプロセッサ301に達する。拡張演算器の制御回路310は、タスク管理部361が出力するタスク1により、拡張演算器のドライバ355を開けることにより、拡張処理部211のもう一方の処理結果データを拡張演算器のデータ処理部出力バッファ332に伝達する。拡張演算器のデータ処理部出力バッファ332は、もう一方の処理結果データを保持する。

【0149】時刻3から時刻4に変化するとき、プロセッサの制御回路305は、EXPIからWBIへパイプラインステージが変化するとき、プロセッサのレジスタライトバッファ116を開け、その結果、プロセッサと拡張演算器とのデータ処理部出力バス125から、プロセッサの汎用レジスタ106へ、ラッチを開いて一方の処理結果データを伝達する。

【0150】時刻4において、プロセッサの汎用レジスタ106は、一方の処理結果データを保持する。

(3. 4)タスク管理部361がタスク2を保持している場合に、データ処理部出力バッファ333が保持するデータをプロセッサの汎用レジスタ106へ書き込む拡張転送命令の動作について説明する。図7に示したパイプライン処理の状態を表すタイムチャートは、本動作においても同じタイミングであるので、図7を参照しつつ説明する。

【0151】時刻1において、プロセッサの命令フェッチステージでは、メモリ103は、命令をメモリからプロセッサと拡張演算器とへの命令供給バス126に載せる。時刻1から時刻2に変化するとき、プロセッサの制御回路305は、IFIからDPIへ移行するため、プロセッサの命令レジスタ112を開ける。拡張演算器の制御回路310は、IFIからDEIへ移行するため、拡張演算器の命令レジスタ119を開ける。

【0152】時刻2において、プロセッサの命令解読ステージ(DPI)では、プロセッサのデコーダ304は、プロセッサの命令レジスタ112が開いたとき、命令を受け取り、命令を解読する。拡張演算器の命令解読ステージ(DEI)では、拡張演算器のデコーダ309は、拡張演算器の命令レジスタ119が開いたとき、命令を受け取り、命令を解読する。

【0153】時刻2から時刻3に変化するとき、プロセッサの制御回路305は、DPIからEXPIへパイプライン

ステージが変化するとき、プロセッサのマイクロ命令レジスタ113、314を開け、その結果、プロセッサのデコーダ304から、プロセッサの制御回路305へ、ラッチを開いてマイクロ命令を伝達し、パイプライン制御を行うプロセッサの制御回路305の制御により、プロセッサと拡張演算器とのデータ処理部入力バス124から、プロセッサのデータ処理部207へ、ラッチを開いてオペランドを伝達する。拡張演算器の制御回路310は、DEIからEXEIへパイプラインステージが変化するとき、拡張演算器のマイクロ命令レジスタ120を開ける。プロセッサの制御回路305は、プロセッサのマイクロ命令レジスタ113が開いたとき、プロセッサのデコーダ304のDPI命令解読ステージが解読したマイクロ命令を受け取る。拡張演算器の制御回路310は、拡張演算器のマイクロ命令レジスタ120が開いたとき、拡張演算器のデコーダ309のDEI命令解読ステージが解読したマイクロ命令を受け取る。

【0154】時刻3において、プロセッサの命令実行ステージ(EXPI)では、プロセッサのデータ処理部207は、データ処理を行わない。拡張演算器の命令実行ステージ(EXEI)では、タスク管理部361は、保持するタスク情報としてタスク2を、拡張演算器の制御回路310へ出力する。拡張演算器の制御回路310は、タスク管理部361が出力するタスク2により、拡張演算器のドライバ358を開けることにより、拡張演算器のデータ処理部出力バッファ333が保持する処理結果データを、プロセッサと拡張演算器とのデータ処理部出力バス125に伝達する。

【0155】時刻2から時刻3に変化するとき、プロセッサの制御回路305は、EXPIからWBIへパイプラインステージが変化するとき、プロセッサのレジスタライトバッファ116を開け、その結果、プロセッサと拡張演算器とのデータ処理部出力バス125から、プロセッサの汎用レジスタ106へ、ラッチを開いて一方の処理結果データを伝達する。

【0156】時刻4において、プロセッサの汎用レジスタ106は、一方の処理結果データを保持する。なお、本実施例においてタスク管理部361がタスクを切り替えることを制御するために、ここでは拡張演算器の制御回路310が制御を行ったが、外部よりタスク切り替え信号線を設け、このタスク切り替え信号によってタスクを切り替えてもよい。

【0157】また、上記第1～第3実施例では、拡張命令のオペランドデータとしてレジスタが指定されている場合を説明したが、オペランドデータとしてメモリデータが指定されている場合も同様に動作することができる。この場合、プロセッサがメモリアクセスを行うので、実行速度の点では従来技術と同等であるが、拡張演算器内部にメモリアクセスを制御する回路が不要になるので、回路規模の点では従来技術よりも小さくすること

ができ、その分コストも低減できる。

【0158】

【発明の効果】以上説明したように、請求項1の発明によれば、拡張処理装置がデータ処理装置のデータ保持手段のデータを用いて演算する場合、及び、拡張処理装置の演算結果をデータ処理装置のレジスタに格納する場合に、一旦メモリを経由してデータ転送する必要がなくなり、データ処理装置と拡張処理装置との間で直接データの受け渡しをするので、処理能力の高い情報処理装置を実現することができるという効果がある。

【0159】請求項2の発明によれば、請求項1の効果に加えて、複数の拡張処理装置を同時に接続して、多様な拡張演算命令が混在するプログラムを高速に実行することができるという効果がある。また、必要とされる拡張処理装置だけを実装することにより、コスト上昇と消費電力の増大とを抑止することができる。さらに拡張処理装置を必要としない場合には、データ処理装置から切り離して実装しなくてもよい。切り離した場合、プロセッサ内部の各配線にかかる電気的負荷が変化しないので切り離しに関して制約を受けないという効果がある。

【0160】請求項3の発明によれば、請求項1の効果に加えて、複数のフラグもオペランドデータと同様に、データ処理装置と拡張処理装置との間で直接受け渡しをするので、より処理能力の高い情報処理装置を実現することができるという効果がある。請求項4の発明によれば、請求項2の効果に加えて、拡張演算実行ステージとデータ処理装置の実行ステージとで、実際のステージのサイクル数を合わせることで、データ処理装置のパイプライン動作を乱すことなく、拡張演算命令の実行と、データ処理装置の命令実行との移行をスムーズに行うことができるという効果がある。

【0161】請求項5の発明によれば、請求項4の効果に加えて、前記実行終了通知を伝達する信号線を追加することにより、パイプライン動作を乱さず、簡単な構成で拡張演算命令への移行をスムーズに行うことができるという効果がある。請求項6の発明によれば、請求項4の効果に加えて、データ処理装置と拡張処理装置との間に拡張命令実行終了を通知する信号線を追加することなく、また、複数の拡張処理装置の接続にも好適した構成が実現できるという効果がある。

【0162】請求項7の発明によれば、請求項1、3または4の効果に加えて、3つ以上のオペランドを要する拡張演算命令の実行に好適した情報処理装置を実現することができるという効果がある。また、複数のオペランドデータをバスを設けて供給する場合と比べて、チップ面積の増大によるコスト上昇と消費電力の増大とを抑止することができる。

【0163】請求項8の発明によれば、請求項7の効果に加えて、拡張演算命令の実行中に割り込み処理が生じた場合、実行時間が増大し性能が劣化を抑止することが

でき、割り込み処理への移行を迅速にすることができるという効果がある。請求項9の発明によれば、請求項7の効果に加えて、拡張演算の結果、複数のデータが生成される場合でも、演算結果をデータ保持手段に高速に格納することができるという効果がある。

【0164】請求項10の発明によれば、請求項9の効果に加えて、割り込み処理への移行を迅速にすることができる、より柔軟な拡張演算をすることができるという効果がある。請求項11の発明によれば、請求項の効果に加えて、複数のタスクを処理する場合に、各タスクに対応して1対の入力バッファと出力バッファとが使用されるので、拡張演算器が処理中にタスク切り換え処理が生じた場合、実行時間の増大と性能劣化とを抑止することができ、高速なタスクスイッチを実現できるという効果がある。

【図面の簡単な説明】

【図1】本発明の第1の実施例における情報処理装置の構成を示すブロック図である。

【図2】同実施例における拡張演算装置に対する拡張命令の一例を示す図である。

【図3】同実施例における情報処理装置におけるパイプライン処理の状態を表すタイムチャートである。

【図4】本発明の第2実施例における情報処理装置の構成を示すブロック図である。

【図5】同実施例の情報処理装置におけるプロセッサの汎用レジスタ106が保持するオペランドをデータ処理部入力バッファ230へ書き込む拡張転送命令のパイプライン処理の状態を表すタイムチャートである。

【図6】同実施例の情報処理装置におけるプロセッサの汎用レジスタ106が保持するオペランドと、データ処理部入力バッファ230が保持するオペランドとを用い、出力結果をプロセッサの汎用レジスタ106とデータ処理部出力バッファ231とへの書き込みを行う拡張処理部211の実行命令のパイプライン処理の状態を表すタイムチャートである。

【図7】本発明の実施例2の情報処理装置におけるデータ処理部出力バッファ231が保持するデータをプロセッサの汎用レジスタ106へ書き込む命令のパイプライン処理の状態を表すタイムチャートである。

【図8】本発明の実施例2の情報処理装置におけるプロセッサ201に割り込みが発生し、拡張演算器のデータ処理部入力バッファ230と、拡張演算器のデータ処理部出力バッファ231とのデータを、退避データ保持バッファ232に格納する動作を表すタイムチャートである。

【図9】本発明の実施例2の情報処理装置における割り込み復帰命令のパイプライン処理の状態を表すタイムチャートである。

【図10】本発明の実施例3における情報処理装置の構成図である。



【図11】本発明の情報処理装置のLSIレイアウトの一例を示す図である。

【符号の説明】

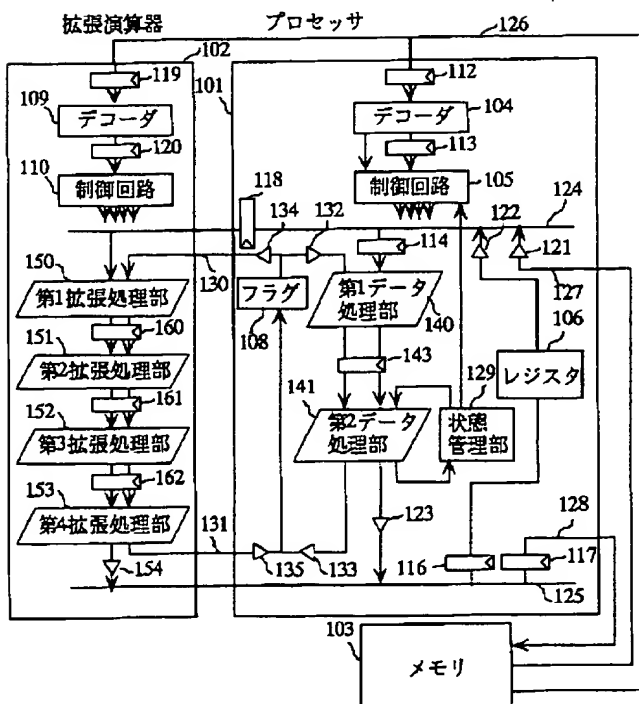
- 101 プロセッサ
- 102 拡張演算器
- 103 メモリ
- 104 デコーダ
- 105 制御回路
- 106 汎用レジスタ
- 108 フラグ格納レジスタ
- 109 デコーダ
- 110 制御回路
- 112 命令レジスタ
- 113 マイクロ命令レジスタ
- 114 データラッチ
- 116 レジスタライトバッファ
- 117 メモリライトバッファ
- 118 拡張演算器用データラッチ
- 119 命令レジスタ
- 120 マイクロ命令レジスタ

- 121 ドライバ
- 121~123 ドライバ
- 124 データ処理部入力バス
- 125 データ処理部出力バス
- 126 命令供給バス
- 127 データバス
- 128 データバス
- 129 状態管理部
- 130 フラグデータバス
- 131 フラグデータバス
- 132~135 ドライバ
- 140 第1データ処理部
- 141 第2データ処理部
- 143 パイプラインラッチ
- 150 第1拡張処理部
- 151 第2拡張処理部
- 152 第3拡張処理部
- 153 第4拡張処理部
- 154 ドライバ

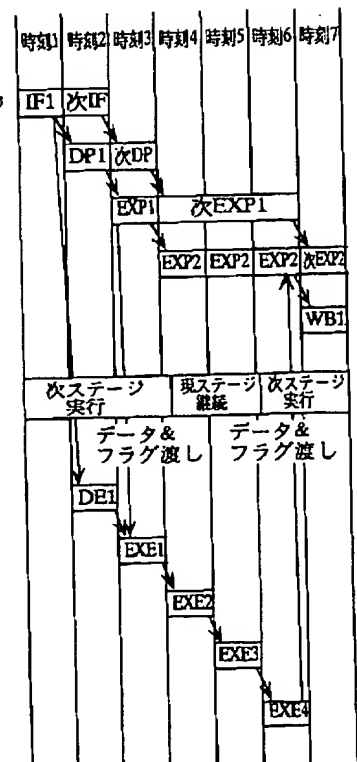
20

【図1】

【図3】



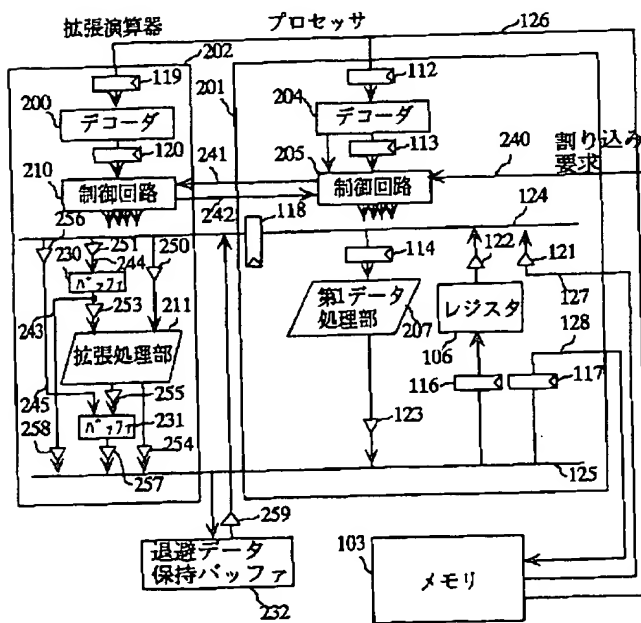
- プロセッサ：命令フェッチステージ
- プロセッサ：命令解読ステージ
- プロセッサ：第1実行ステージ
- プロセッサ：第2実行ステージ
- プロセッサ：書き込みステージ
- プロセッサ：状態管理部の状態
- 拡張演算器：命令解読ステージ
- 拡張演算器：第1実行ステージ
- 拡張演算器：第2実行ステージ
- 拡張演算器：第3実行ステージ
- 拡張演算器：第4実行ステージ



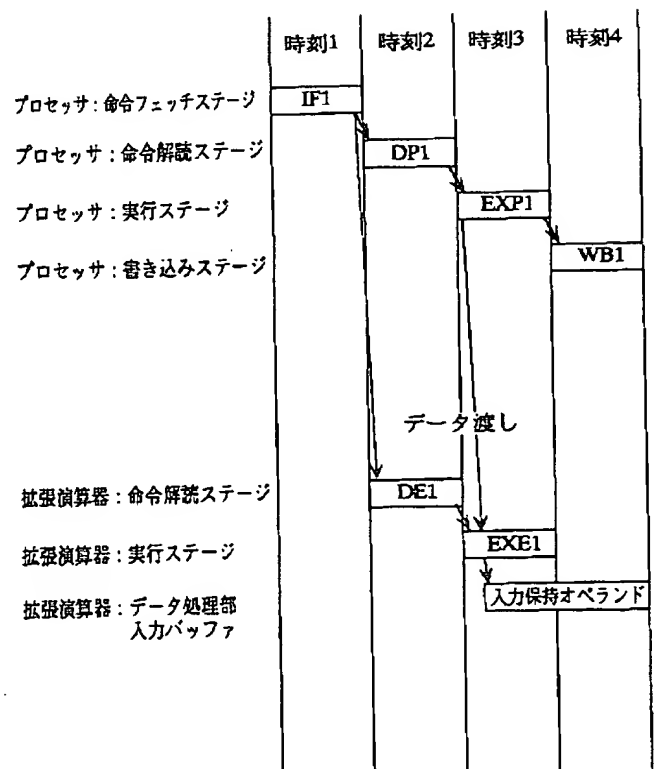
【図 2】

ニーモニック	命令コード	動作
MULQ Dm,Dn	F60X(H) ↑ ソース、ディスチネーション	$Dm \times Dn$ → Dn(lower32),MR
MULQ imm8,Dn	F90XXX(H) ↑ ↑ imm8 ↑ ソース、ディスチネーション	(imm8の符号拡張) $\times Dn$ → Dn(lower32),MR(upper32)
MULQ imm16,Dn	FB0XXXXX(H) ↑ ↑ imm16 ↑ ソース、ディスチネーション	(imm16の符号拡張) $\times Dn$ → Dn(lower32),MR(upper32)
MULQ imm32,Dn	FD0XXXXXXXXX(H) ↑ ↑ imm32 ↑ ソース、ディスチネーション	(imm32の符号拡張) $\times Dn$ → Dn(lower32),MR(upper32)

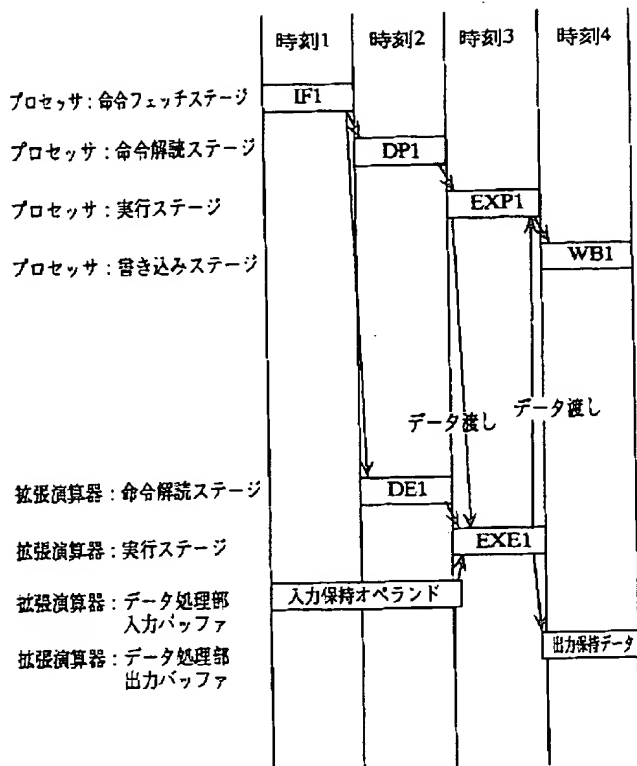
【図 4】



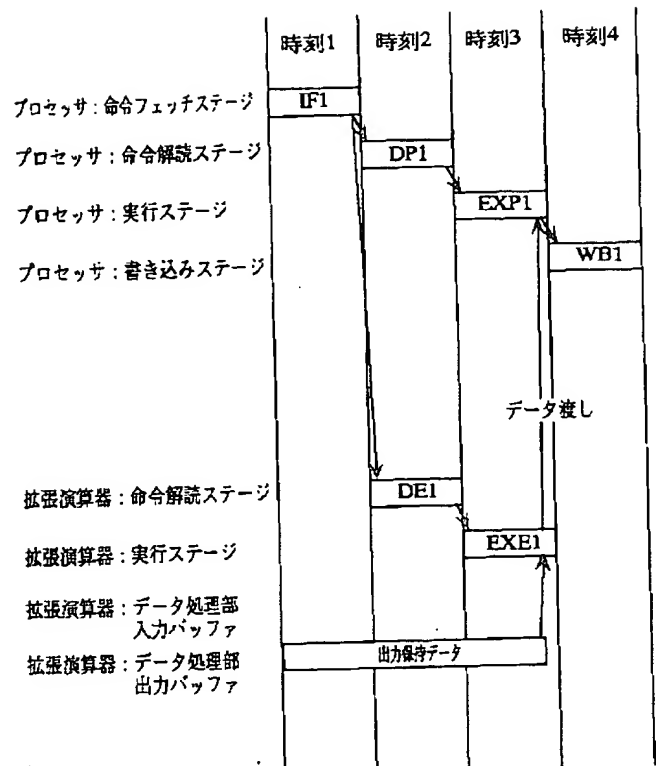
【図 5】



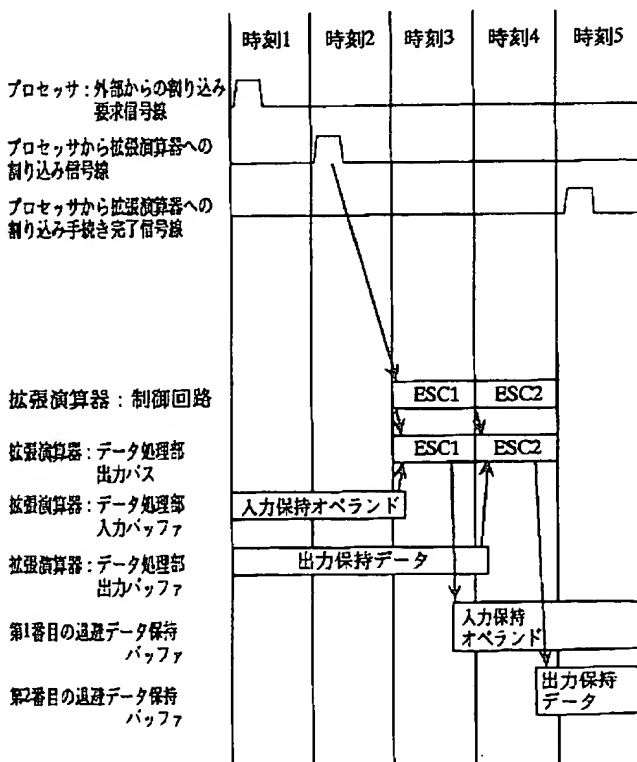
【図 6】



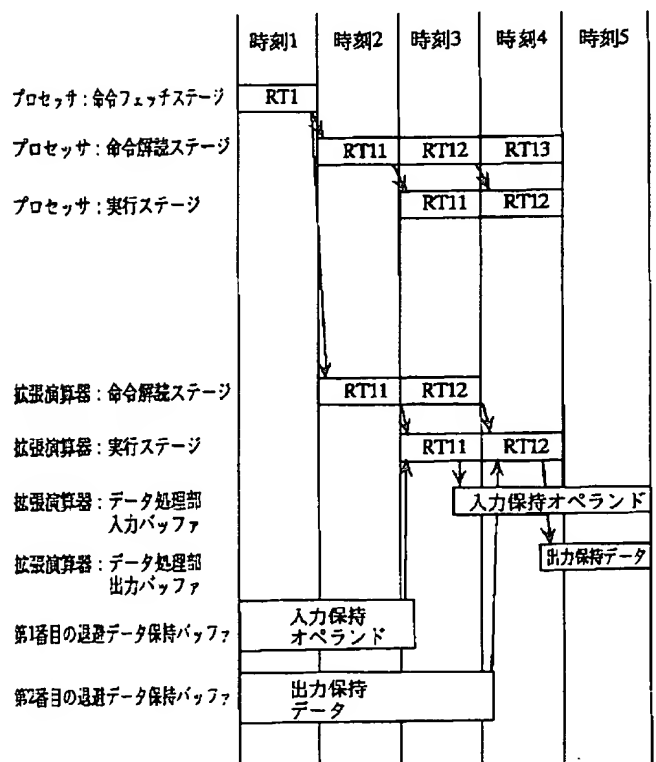
【図 7】



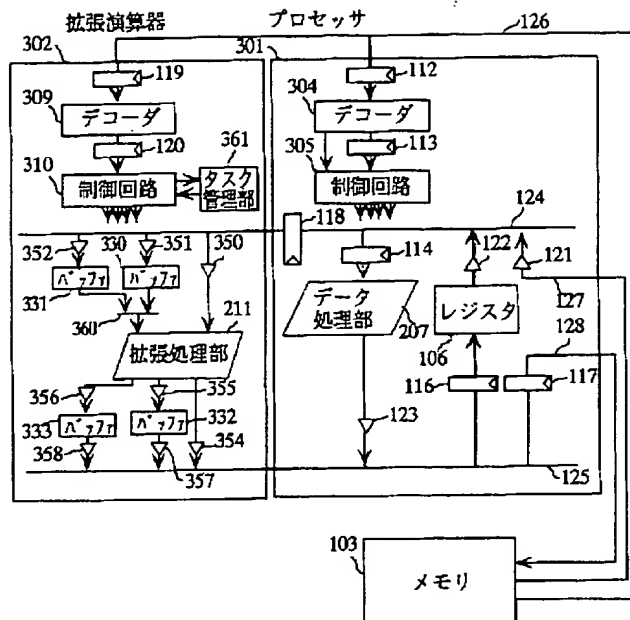
【図 8】



【図 9】



【図 10】



【図 11】

レイアウト概念図

